

Shazia Sadiq *Editor*

# Handbook of Data Quality

*Research and Practice*

# **Handbook of Data Quality**



Shazia Sadiq  
Editor

# Handbook of Data Quality

## Research and Practice



Springer

*Editor*  
Shazia Sadiq  
University of Queensland  
Brisbane  
Australia

ISBN 978-3-642-36256-9      ISBN 978-3-642-36257-6 (eBook)  
DOI 10.1007/978-3-642-36257-6  
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013935616

ACM Computing Classification (1998): H.2, H.3, E.5, K.6

© Springer-Verlag Berlin Heidelberg 2013

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media ([www.springer.com](http://www.springer.com))

# Advisory Panel

**Carlo Batini** Università degli Studi di Milano - Bicocca, Milano, Italy

**Yang Lee** North Eastern University, Boston, MA, USA

**Chen Li** University of California Irvine, Irvine, CA, USA

**Tamer Ozsu** University of Waterloo, Waterloo, ON, Canada

**Felix Naumann** Hasso Plattner Institute, Potsdam, Germany

**Barbara Pernici** Politecnico di Milano, Milano, Italy

**Thomas Redman** Navesink Consulting Group, Rumson, NJ, USA

**Divesh Srivastava** AT&T Labs-Research, Florham Park, NJ, USA

**John Talburt** University of Arkansas at Little Rock, Little Rock, AR, USA

**Xiaofang Zhou** The University of Queensland, Brisbane, Australia



*To my late father, Aziz Beg  
historian, author and diplomat*



# Preface

The impact of data quality on the information chain has been widely recognized since the onset of large-scale data processing. Furthermore, recent years have seen a remarkable change in the nature and usage of data itself due to the sheer volume of data, high accessibility leading to unprecedented distribution and sharing of data, and lack of match between the intention of data creation and its subsequent usage, to name a few. The importance of the understanding and management of data quality for individuals, groups, organizations, and government has thus increased multifold.

The data (and information) quality domain is supported by several decades of high quality research contributions and commercial innovations. Research and practice in data and information quality is characterized by methodological as well as topical diversities. The cross-disciplinary nature of data quality problems as well as a strong focus on solutions based on the fitness for use principle has further diversified the related body of knowledge. Although research pluralism is highly warranted, there is evidence that substantial developments in the past have been isolationist. As data quality increases in importance and complexity, there is a need to motivate exploitation of synergies across diverse research communities.

The above factors warrant a multipronged approach to the study of data quality management spanning: organizational aspects, i.e. strategies to establish people, processes, policies, and standards required to manage data quality objectives; architectural aspects, i.e. the technology landscape required to deploy developed processes, standards, and policies; and computational aspects which relate to effective and efficient tools and techniques for data quality.

Despite a significant body of knowledge on data quality, the community is lacking a resource that provides a consolidated coverage of data quality over the three different aspects. This gap motivated me to assemble a point of reference that reflects the full scope of data quality research and practice.

In the first chapter of this handbook, I provide a detailed analysis of the data quality body of knowledge and present the rationale and approach for this handbook, particularly highlighting the need for cross-fertilization within and across research and practitioner communities. This handbook is then accordingly structured into three parts representing contributions on organizational, architectural,

and computational aspects. There is also a fourth part, devoted to case studies of successful data quality initiatives that highlight the various aspects of data quality in action. This handbook concludes with a chapter that outlines the emerging data quality profession, which is particularly important in light of new developments such as big data, advanced analytics, and data science.

The preparation of this handbook was undertaken in three steps. Firstly, a number of global thought leaders in the area of data quality research and practice were approached to join the initiative as part of the advisory panel. The panel members contributed significantly to the refinement of this handbook structure and identification of suitable chapter authors and also supported the review process that followed chapter submissions. The identified chapter authors were then invited to provide contributions on the relevant topics. Finally, all chapter contributions were reviewed by at least two experts. To ensure that the quality of the final chapters was not compromised in any way, some contributions were rejected or substantially revised over two or three review cycles. However, I am most grateful for the time devoted by all authors to produce high quality contributions and especially for the responsiveness of the authors towards making the required changes.

I would like to take this opportunity to thank all the authors and advisors for their valuable contributions. A special thanks to Xiaofang Zhou, Divesh Srivastava, Felix Naumann, and Carlo Batini for their guidance and inspiration in the preparation of this handbook. Thanks to all the expert reviewers of the chapters, with a special thanks to Mohamed Sharaf for constant encouragement and advice. Last but not least, a big thanks to Kathleen Williamson, Yang Yang and Vimukthi Jayawardene for an enormous help in the editing and formatting work required for the preparation of this handbook.

I hope that this Handbook of Data Quality will provide an appreciation of the full scope and diversity of the data quality body of knowledge and will continue to serve as a point of reference for students, researchers, practitioners, and professionals in this exciting and evolving area.

Brisbane, Australia  
April 2013

Shazia Sadiq

# Contents

<b>Prologue: Research and Practice in Data Quality Management .....</b>	1
Shazia Sadiq	

## **Part I Organizational Aspects of Data Quality**

<b>Data Quality Management Past, Present, and Future: Towards a Management System for Data .....</b>	<b>15</b>
Thomas C. Redman	
<b>Data Quality Projects and Programs .....</b>	<b>41</b>
Danette McGilvray	
<b>Cost and Value Management for Data Quality .....</b>	<b>75</b>
Mouzhi Ge and Markus Helfert	
<b>On the Evolution of Data Governance in Firms: The Case of Johnson &amp; Johnson Consumer Products North America .....</b>	<b>93</b>
Boris Otto	

## **Part II Architectural Aspects of Data Quality**

<b>Data Warehouse Quality: Summary and Outlook .....</b>	<b>121</b>
Lukasz Golab	
<b>Using Semantic Web Technologies for Data Quality Management .....</b>	<b>141</b>
Christian Fürber and Martin Hepp	
<b>Data Glitches: Monsters in Your Data .....</b>	<b>163</b>
Tamraparni Dasu	

**Part III Computational Aspects of Data Quality**

<b>Generic and Declarative Approaches to Data Quality Management .....</b>	181
Leopoldo Bertossi and Loreto Bravo	
<b>Linking Records in Complex Context .....</b>	213
Pei Li and Andrea Maurino	
<b>A Practical Guide to Entity Resolution with OYSTER .....</b>	235
John R. Talburt and Yinle Zhou	
<b>Managing Quality of Probabilistic Databases .....</b>	271
Reynold Cheng	
<b>Data Fusion: Resolving Conflicts from Multiple Sources.....</b>	293
Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava	

**Part IV Data Quality in Action**

<b>Ensuring the Quality of Health Information: The Canadian Experience .....</b>	321
Heather Richards and Nancy White	
<b>Shell's Global Data Quality Journey .....</b>	347
Ken Self	
<b>Creating an Information-Centric Organisation Culture at SBI General Insurance .....</b>	369
Ram Kumar and Robert Logie	
<b>Epilogue: The Data Quality Profession .....</b>	397
Elizabeth Pierce, John Talburt, and C. Lwanga Yonke	
<b>About the Authors .....</b>	419
<b>Index .....</b>	431

# Prologue: Research and Practice in Data Quality Management

Shazia Sadiq

**Abstract** This handbook is motivated by the presence of diverse communities within the area of data quality management, which have individually contributed a wealth of knowledge on data quality research and practice. The chapter presents a snapshot of these contributions from both research and practice, and highlights the background and rational for the handbook.

## 1 Introduction

Deployment of IT solutions, often following from strategic redirections, upgrades, mergers and acquisitions, is inevitably subjected to an evaluation of return on investment (ROI), which includes evaluation of the costs of sizable installations as well as the cost of changing the culture and work practice of all involved. It is often observed that the results of such analyses frequently indicate a failure to achieve the expected benefits [2]. A range of factors contributes to dismal ROIs, including significant factors rooted externally to the technological sophistication of the systems and often residing in the quality of the information the system manages and generates.

The issue of data quality is as old as data itself. However, it is now exposed at a much more strategic level, e.g. through business intelligence (BI) systems, increasing manifold the stakes involved for corporations as well as government agencies. For example, the Detroit terror case triggered an overhaul of the nationwide watch list system, where lack of data propagation/consistency and issues with data freshness can be observed. The issue is equally important for scientific applications where lack of knowledge about data accuracy, currency or certainty can lead to catastrophic results. For example, the hurricane protection system in

---

S. Sadiq (✉)  
The University of Queensland, Brisbane, Australia  
e-mail: [shazia@itee.uq.edu.au](mailto:shazia@itee.uq.edu.au)

New Orleans failed because it was “inadequate and incomplete”, having been built disjointedly over several decades using outdated elevation data (New York Times, June 1, 2006). Further, the proliferation of shared/public data as on the World Wide Web and growth of the Web community has increased the risk of poor data quality usage for individuals as well. This is particularly alarming due to the diversity of the Web community, where many are unaware of data sources and data credentials. The situation is further complicated by presence of data aggregations and assimilations, e.g. through meta-search engines where source attribution and data provenance can be completely hidden from the data consumers.

One can also observe the changing nature of data quality management over the last decade or more. First, there are clear implications that relate to the sheer volume of data produced by organizations today. Second, recent years have seen an increase in the diversity of data. Such diversity refers to structured, unstructured and semi-structured data and multimedia data such as video, maps and images. Data also has an increasing number of sources. The use of various technologies, for example, sensor devices, medical instrumentation and RFID readers, further increases the amount and diversity of data being collected. More subtle factors also exist—such as the lack of clear alignment between the intention of data creation and its subsequent usage. A prime example of such lack of alignment is the vast amount of data collected from social networks that can then be used, without assessment of quality, as a basis for design and marketing decisions. Accordingly, a related factor exists that relates to difficulties in defining appropriate data quality metrics.

As these changes occur, traditional approaches and solutions to data management in general, and data quality control specifically, are challenged. There is an evident need to incorporate data quality considerations into the whole data cycle, encompassing managerial/governance as well as technical aspects. Currently, data quality contributions from research and industry appear to originate from three distinct communities:

Business analysts, who focus on *organizational* solutions. That is, the development of data quality objectives for the organization as well as the development of strategies to establish roles, processes, policies and standards required to manage and ensure the data quality objectives are met.

Solution architects, who work on *architectural* solutions. That is, the technology landscape required to deploy developed data quality management processes, standards and policies.

Database experts and statisticians, who contribute to *computational* solutions. That is, effective and efficient IT tools, and computational techniques, required to meet data quality objectives. Techniques in this regard can include record linkage, lineage and provenance, data uncertainty, semantic integrity constraints as well as information trust and credibility.

For the research community to adequately respond to the current and changing landscape of data quality challenges, a unified framework for data quality research is needed. Such a framework should acknowledge the central role of data quality in future systems development initiatives and motivate the exploitation of synergies across diverse research communities. It is unclear if synergies across the three contributing communities have been fully exploited. The sections below substantiate

this observation through an analysis of last 20 years of literature on data quality [14]. We argue that a unified framework for data quality management should bring together organizational, architectural and computational approaches proposed from the three communities, respectively.

## 2 Related Studies

A number of studies have addressed the issue of defining and analysing the scope of data quality research in the past. Owing to the cross-disciplinary needs of this area, identifying the central themes and topics and correspondingly the associated methodologies has been a challenge. In [10], a framework is presented that characterizes data quality research along the two dimensions of topics and methods, thereby providing a means to classify various research works. Previous works have also assisted by developing frameworks through which data quality research could be characterized, including a predecessor framework by the above group [17] that analogized data quality processes with product manufacturing processes. Some key research aspects such as data quality standardization, metrics/measurements and policy management emerged from these earlier works.

Other more recent studies have also provided valuable means of classification for data quality research. Ge and Helfert [5] have structured their review of the literature as IQ Assessment, IQ Management and Contextual IQ. Lima et al. [8] classify the literature between theoretical (conceptual, applied, illustrative) and practical (qualitative, experimental, survey, simulation) aspects. Neely and Cook [12] present their classification as a cross tabulation of Wang's framework [17] and Juran's original fitness for use factors [7].

The above studies provide various angles through which the body of knowledge can be classified and thus provide an essential means of understanding the core topics of data quality. However, understanding the intellectual corpus of a discipline requires not only an understanding of its core but also its boundaries [1]. As the realm of data quality has grown, so has the scope of its reference disciplines. With these factors in mind, we focused our study on understanding the interconnections and synergies across the various communities that contribute to data quality, rather than an identification of its central themes. The sections below substantiate this observation through an analysis of last 20 years of literature on data quality [14]. We argue that addressing the current challenges in data quality warrants such an understanding so synergies would be better exploited and holistic solutions may be developed.

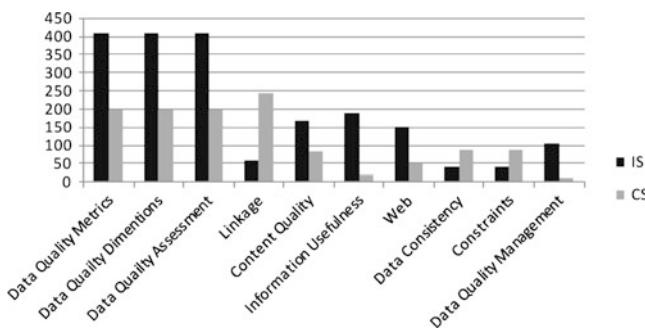
## 3 Results of Literature Analysis

As a first step towards understanding the gaps between the various research communities, we undertook a comprehensive literature study of data quality research published in the last two decades [14]. In this study we considered a broad range of

**Table 1** Considered publication outlets

	<b>Includes<sup>a</sup></b>	<b>Total</b>
CS Conferences	BPM, CIKM, DASFAA, ECOOP, EDBT, PODS, SIGIR, SIGMOD, VLDB, WIDM, WISE	7,535
IS Conferences	ACIS, AMCIS, CAiSE, ECIS, ER, HICSS, ICIQ, ICIS, IFIP, IRMA, PACIS	13,256
CS Journals	TODS, TOIS, CACM, DKE, DSS, ISJ (Elsevier), JDM, TKDE, VLDB Journal	8,417
IS Journals	BPM, CAIS, EJIS, Information and Management, ISF, ISJ (Blackwell), ISJ (Sarasota), JMIS, JAIS, JISR, MISQ, MISQ Executive	2,493

<sup>a</sup>Due to space limitation, widely accepted abbreviations have been used, where full names are easily searchable via WWW

**Fig. 1** Keyword frequency between IS and CS outlets

Information System (IS) and Computer Science (CS) publication (conference and journal) outlets (1990–2010) so as to ensure adequate coverage of organizational, architectural and computational contributions (see Table 1).

The main aims of the study were to understand the current landscape of data quality research, to create better awareness of (lack of) synergies between various research communities and, subsequently, to direct attention towards holistic solutions that span across the organizational, architectural and computational aspects (thus requiring collaboration from the relevant research communities).

In this section we present brief excerpts of the literature analysis conducted in [14] and [15] to provide a snapshot of the current research landscape in data quality management. As a consequence of the above studies, from the original data set of over 30,000 articles, a bibliographical database of almost 1500 publications (together with related keywords) was created through a rigorous analytical and reproducible methodology as detailed in [14].

The analysis revealed **topics** and **venues** of highest frequency as shown in Fig. 1 and Table 2, respectively. From the above, there is a clear indication that data quality themes are spread between IS and CS outlets. The overall distribution of papers between IS and CS outlets is summarized in Fig. 1. Clearly there are some topics

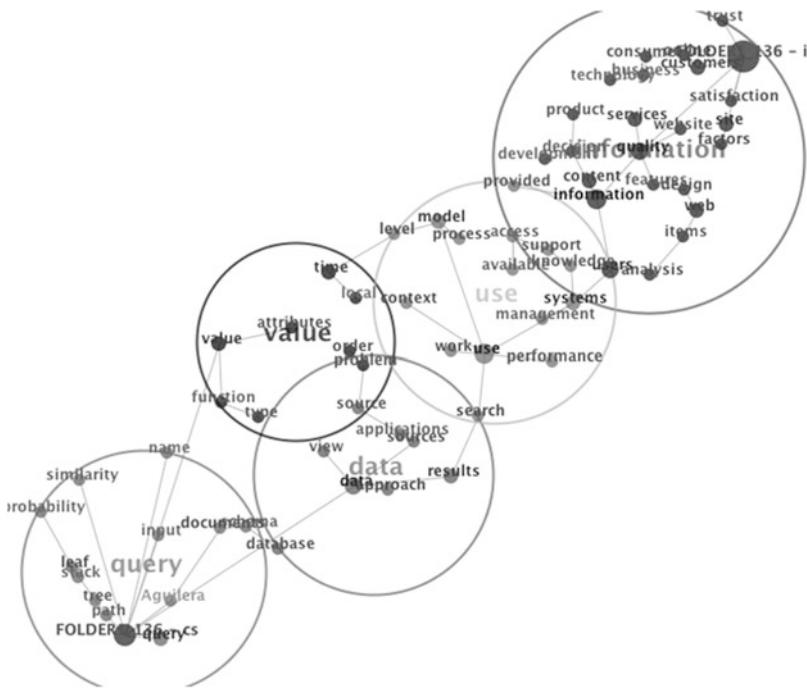
**Table 2** Top publication frequencies with respect to publication venue

Publication outlet	# Pubs
International Conference on Information Quality (ICIQ)	241
Americas Conference on Information Systems (AMCIS)	152
International Conference on Very Large Databases (VLDB)	148
IEEE Transactions on Knowledge and Data Engineering (DKE)	120
ACM SIGMOD International Conference on Management of Data (SIGMOD)	116
ACM Transactions on Information Systems (TOIS)	51
Communication of the ACM (CACM)	49
Pacific Asia Conference on Information Systems (PACIS)	45
Hawaii International Conference on System Sciences (HICSS)	44
Symposium on Principles of Database Systems (PODS)	36
ACM Transactions on Database Systems (TODS)	35
International conference on Information Systems (ICIS)	34
European Conference on Information Systems (ECIS)	33
Australasian conference on Information Systems (ACIS)	33
Journal of Information & Management (IM)	27
ACM Special Interest Group on Information Retrieval (SIGIR)	27
International Conference on Extending Database Technology (EDBT)	22
International Conference on Database Systems for Advanced Applications (DASFAA)	20
Journal of Management Information Systems (MIS)	19
International Workshop on Information Quality in Information Systems (IQIS)	18
Journal of Information Systems Research (ISR)	12
Management Information Systems Quarterly (MISQ)	12
International Conference on Advanced Information Systems Engineering (CAISE)	10

where the overlap is greater (e.g. *Data Quality Metrics*) than others (e.g. *Linkage* and *Information Usefulness*).

Table 2 provides an alternative view for observing research activity in relation to prominent IS and CS publication venues. Obviously the International Conference on Information Quality (ICIQ) has the highest number of publications that span across a large number of keywords, with *Data Quality Assessment*, *Metrics* and *Dimensions* being the dominant ones. For AMCIS, in addition to the above keywords, *Information Usefulness* and *Content Quality* were also observed. Similarly, for VLDB as well as DKE journal, *Linkage* was the dominant keyword, closely followed by *Data Consistency* and *data Uncertainty*.

We further conducted a **thematic** analysis of the papers through a text-mining tool called Leximancer ([www.leximancer.com](http://www.leximancer.com)). Leximancer performs a full text analysis both systematically and graphically by creating a map of the concepts and themes reappearing in the text. The tool uses a machine-learning technique based on a Bayesian approach to prediction. Leximancer uses concept maps to visualize the relationships. Each of the identified concepts is placed on the map in proximity of other concepts in the map through a derived combination of the direct and indirect relationships between those concepts (see Fig. 2). Concepts are represented by labelled and colour-coded dots. The size and brightness of a dot representing a



**Fig. 2** Content quality—Information Systems vs. Computer Science publication focus

given concept on the map, is indicative of the concept's strength within the body of analysed text. The thickness and the brightness of connections between concepts are indicative of the frequency of co-occurrence of the two concepts. The relative distance of concepts on the map is indicative of similar conceptual contexts—i.e. the shorter the distance between the two concepts, the closer in context they are. Thematic clusters of highly connected concepts are indicated through coloured circles, called themes.

To explore synergies and differences between data quality research in the CS and IS disciplines, we conducted a series of Leximancer analyses for each of the top 10 keywords listed in Fig. 1. For each of the keywords, data was analysed considering the CS publications in isolation, then considering the IS publications in isolation, followed by a joint analysis of both data sets to gain a better understanding of the common focus of the two disciplines.

Due to space limitations, a detailed analysis is omitted here. However, as an example consider Fig. 2, where we selected the set of “Content Quality”-related publications in the CS and IS publication outlets for a joint analysis. Here the collective Information Systems data set related to the “Content Quality” topic is indicated by a “FOLDER-136-IS” concept. Likewise, the Computer Science data set is represented by “FOLDER-136-CS” concept. Specifically, it shows the relationships of concepts related to Content Quality across all considered

**Table 3** Authors with more than 1,000 citations

Author	Citations	Author	Citations
Wang, R. Y.	4,364	McLean, E. R.	1,373
Widom, J.	2,774	Halevy, A.	1,308
Strong, D.	1,986	Lenzerini, M.	1,299
Ng, R. T.	1,894	Lee, Y. W.	1,183
Motwani, R.	1,847	Gibbons, P. B.	1,105
Datar, M.	1,739	Knorr, E. M.	1,071
Babcock, B.	1,685	Koudas, N.	1,061
Babu, S.	1,607	Chaudhuri, S.	1,056
Garofalakis, M. N.	1,428	Shim, K.	1,051
Rastogi, R.	1,378	Hellerstein, J. M.	1,014
DeLone, W.	1,373		

publication years and how the data set relates to concepts that were identified to be the strongest common concepts across the two data sets. Our analysis indicates that, while there are concepts that are common to both data sets, the strength of the connection is weak (while this is not visible in Fig. 2, due to resolution, the weakness is indicated in the Leximancer tool environment by the relative lack of thick, bright connections between both folder concepts and any one of the Content Quality concepts).

Indeed, the analysis uncovers strong evidence that the Information Systems set of papers is strongly focused on information quality, issues relating to satisfaction and business value in general, yet it is not as strongly focused (as indicated by the relative distance of the themes from each other and the relative closeness of the themes to each of the two publication sets) on approaches for ensuring content quality. While this is not surprising in itself, given that Information Systems is less technically oriented, we see a weakness in a situation where the communities that should be collaborating together appear to lack a strong collaboration and common focus.

We also conducted a *citation* analysis. For this purpose, we wrote a crawler script that searches all papers in the database within Google scholar and collects information regarding number of citations for the paper. In Table 3 we list the top cited authors. It is important to note that the citation counts are entirely based on the publications which are part of our collection and thus do not reflect the overall count for authors.

Some of the earliest contributions came from Wang, R. Y., Strong, D. and associates on the identification of *Data Quality Dimensions* and *Data Quality Assessment*. These contributions have been heavily utilized by later researchers as is evident from the high citation count above. Widom, J. and co-authors have contributed substantially to the body of knowledge on *data lineage* and *uncertainty* especially through the Trio system (see infolab.stanford.edu/trio). Similarly works of Ng, R. T. on identification of outliers in large data sets have applications in *error detection*, *entity resolution* and a number of data quality-related problems. Although it is not possible to summarize the contributions of all highly cited authors, it is safe to conclude that the contributions of these influential contributors are indicative of the wide span of data quality research.

## 4 The Three Pillars of Data Quality Management

The diversity and span of data quality research evident from the above-presented analysis of research literature from CS and IS publications is further exaggerated when we consider the vast experiential knowledge found in the practitioner and professional community within the information industry. Data quality management has been supported for last several decades by a number of highly active and experienced practitioners, including but not limited to [3, 9, 11, 13].

There have also been some industry-led initiatives that have attempted to identify key requirements or demands from industry in terms of data quality management [6]. The most relevant and recent of which is a job analysis report published by the International Association for Information and Data Quality ([iaidq.org](http://www.iaidq.org)). The report provides data that assists in understanding and establishing the roles of data quality professionals in industry. Additionally, the report also identifies the body of knowledge required by those professionals to provide information/data quality services across various roles of an organization [18].

The contributions from various industry sources as above are inclined towards the *organizational* aspects of data quality management. For example, the industry-driven Information Quality Certification Program ([www.iaidq.org/iqcp](http://www.iaidq.org/iqcp)) covers six domains of (1) Information Quality Strategy and Governance, (2) Information Quality Environment and Culture, (3) Information Quality Value and Business Impact, (4) Information Architecture Quality, (5) Information Quality Measurement and Improvement and (6) Sustaining Information Quality. Although the organizational issues are an essential aspect of the overall space for data quality, it is also evident that lack of appropriate tools and systems to support organizational initiatives on data quality will undermine the best efforts of a dedicated team. This becomes especially apparent in the presence of large data sets, on one hand, and the complex dynamics of IT systems, enterprise software and legacy applications, on the other. There is a substantial body of knowledge that exists in support of such challenges, such as advanced record linkage, entity resolution, duplicate detection, managing uncertain data and data tracking and lineage. Most of these solutions are based on advanced *computational* techniques.

Finally, to state the obvious, there is a multibillion dollar data management market and commercial products and solutions that provide technology-related products and services across data(base) management, data integration and data analytics (including data warehousing and business intelligence solutions). Many of these vendors provide solutions directly related to data quality management [4]. These solutions provide the space in which many data quality solutions are deployed. Alignment between the organizational objectives and the technology *architecture* of deployed solutions is imperative.

In spite of the large body of knowledge stemming from research, practitioner and vendor communities, recent studies of data professionals indicate that a resounding 68 % of data quality problems are still detected due to complaints and/or by chance [16]. We argue that a key contributing issue is the segregated nature of the body of



**Fig. 3** The three pillars of data quality. *Organizational*: the development of data quality objectives for the organization, as well as the development of strategies to establish roles, processes, policies and standards required to manage and ensure the data quality objectives are met. *Architectural*: the technology landscape required to deploy developed data quality management processes, standards and policies. *Computational*: effective and efficient IT tools, and computational techniques, required to meet data quality objectives. Techniques in this regard can include record linkage, lineage and provenance, data uncertainty, semantic integrity constraints, as well as information trust and credibility

knowledge for data quality management and technology solutions. Next-generation solutions need to embrace the diversity of the data quality domain and build upon the three foundation pillars relating to organizational, architectural and computational aspects of data quality as depicted in Fig. 3.

As a simple example to illustrate the necessity of the three pillars, consider the following scenario:

A large distribution company (LDC) that acquires two other distribution establishments, which will now form part of LDC operations as subsidiaries while maintaining their individual brandings. Each of the subsidiaries may have their own partner suppliers along with item catalogs. Consider the case that there is a large overlap of business with a particular supplier group, which may put LDC into a favorable bargaining position to negotiate significant discounts. However, data differences do not reveal this position, and thus directly impact on the bottom line for LDC.

In its simplest form a solution for the above scenario may be:

1. Create a reference (synonym) table for suppliers
2. Load supplier data from all subsidiaries into the reference table
3. Use matching techniques to identify potential overlaps
4. Extract a master table for suppliers—represents a single version of truth
5. Retain original representations—represent multiple versions of truth
6. Allow access for subsidiaries to reference master data in all new (or update) transactions involving supplier data
7. Ensure data managers are accountable for continued master data checks
8. Introduce a periodic monitoring scheme

Steps 1–5 require management intervention but at the same time require computational expertise specifically for step 3 and at the very least IT support for the

**Table 4** Topics covered in the handbook

Prologue: Research and Practice in Data Quality Management			
Part I Organizational	Part II Architectural	Part III Computational	Part IV Data Quality in Action
Data Quality Management: history, frameworks, DQ projects and DQ programs (1, 2)	DQ issues for Data Warehouses (5)	DQ Rules and Constraints (8)	Case Study presenting successful Data Integration (13)
Data Quality Costs (3)	Role of Semantics and Ontologies for DQ (6)	Record Linkage, Duplicate Detection and Entity Resolution (9, 10)	Case Study presenting a longitudinal process-oriented DQ initiative (14)
Governance and Maturity (4)	Data Quality Assessment and Error Detection (7)	Managing Data Uncertainty (11)	Case Study focusing on creating a culture of Information Management (15)
Epilogue: The Data Quality Profession			

other steps. Step 6 demands that the company put in place the requisite technology (systems, networks, etc.) to enable appropriate access mechanisms and transactional consistency. Steps 7–8 are primarily management related although it is clear that previous steps will also need support of management, business teams and data owners.

## 5 Handbook Topics

The rationale for this handbook is motivated by the above analysis of research and practice in data quality management. The handbook is accordingly structured in to three parts representing contributions on organizational, architectural and computational aspects of data quality management, with the last part devoted to case studies of successful data quality initiatives that highlight the various aspects of data quality in action. The book concludes with a chapter that outlines the emerging data quality profession, which is particularly important in light of new developments such as big data, advanced analytics and data science. The four parts of the handbook and constituent topics (chapters) are summarized in Table 4.

Most chapters that focus on specific topics present both an overview of the topic in terms of historical research and/or practice and state of the art as well as specific techniques, methodologies or frameworks developed by the individual contributors.

Researchers and students from Computer Science, Information Systems as well as Business Management can benefit from this text by focusing on various sections relevant to their research area and interests. Similarly data professionals and practitioners will be able to review aspects relevant to their particular work. However, the biggest advantage is expected to emerge from wider readership of chapters that may not be directly relevant for the respective groups.

## References

1. Benbasat I, Zmud RW (2003) The identity crisis within the IS discipline: designing and communicating the discipline's core properties. *MIS Q* 27(2):183–194
2. Carr N (2004) Does IT matter? Information technology and the corrosion of competitive advantage. Harvard Business School Press, Boston
3. English LP (2009) Information quality applied: best practices for improving business information processes and systems. Wiley, Indiana
4. Gartner Magic Quadrant for Data Quality. [http://www.citia.co.uk/content/files/50\\_161-377.pdf](http://www.citia.co.uk/content/files/50_161-377.pdf). Accessed 15 Oct 2012
5. Ge M, Helfert M (1996) A review of information quality research. In: The 12th international conference on information quality. MIT, Cambridge, pp 1–9
6. Harte-Hanks Trillium Software 2005/6 Data Quality Survey (2006) <http://infoimpact.com/Harte-HanksTrilliumSoftwareDQSurvey.pdf>. Accessed 15 Oct 2012
7. Juran JM (1962) Quality control handbook
8. Lima LFR, Macada ACG, Vargas LM (2006) Research into information quality: a study of the state of the art in IQ and its consolidation. In: 11th international conference on information quality. MIT, Cambridge
9. Loshin D (2006) Monitoring data quality performance using data quality metrics. Informatica Corporation, Redwood City
10. Madnick SE, Wang RY, Lee YW, Zhu H (2009) Overview and framework for data and information quality research. *J Data Information Qual* 1(1):1–22
11. McGilvray D (2008) Executing data quality projects: ten steps to quality data and trusted information. Morgan Kaufmann, Burlington
12. Neely MP, Cook J (2008) A framework for classification of the data and information quality literature and preliminary results (1996–2007). *AMCIS Proceedings*. Accessed from [http://www.citia.co.uk/content/files/50\\_161-377.pdf](http://www.citia.co.uk/content/files/50_161-377.pdf). Accessed 15 Oct 2012
13. Redman TC (1996) Data quality for the information age. Artech House, Boston
14. Sadiq S, Yeganeh NK, Indulska M (2011) 20 years of data quality research: themes, trends and synergies. In: Proceedings of the 22nd Australasian Database Conference (ADC 2011), Perth, WA, Australia. 17–20 January 2011, pp 1–10
15. Sadiq S, Yeganeh NY, Indulska M (2011) An analysis of cross-disciplinary collaborations in data quality research. In: European conference on information systems (ECIS 2011), Helsinki, Finland, 2011
16. Sadiq S, Jayawardene V, Indulska M (2011) Research and industry synergies in data quality management. In: International conference on information quality (ICIQ2011), Adelaide, Australia, 18–20 November, 2011
17. Wang RY, Storey VC, Firth CP (1995) A framework for analysis of data quality research. *IEEE Trans Knowledge Data Eng* 7(4):623–640
18. Yonke CL, Walenta C, Talburt JR (2011) The job of the information/data quality professional. International Association for Information and Data Quality. Available from <http://iaidq.org/publications/yonke-2011-02.shtml>. Accessed 15 Oct 2012

# **Part I**

## **Organizational Aspects of Data Quality**

This part covers the organizational aspect of data quality management that relates to the development of data quality objectives for the organization and strategies to establish the people, processes, policies, and standards required to manage and ensure the data quality objectives are met. Over the last several decades, the research and practitioner community has contributed significantly to the development of a range of management approaches resulting in widely used frameworks and processes, and contributing to improved understanding of data governance and maturity models. This part presents four contributions on key topics relating to the organizational aspect.

The first chapter in this part by Thomas Redman aka “Data Doc” provides a detailed review of the past, present, and future of data quality management bringing over two decades of the author’s field experience to bear. This chapter also outlines and directs attention of research efforts towards a new systems approach for management of data and data quality.

The second chapter provides practical guidelines for initiation and sustainability of data quality initiatives spanning both short-term projects and long-term programs that can systematically move organizations towards data governance maturity. The chapter provided by Danette McGilvray is based on the Ten Steps to Quality Data and Trusted Information™ methodology that the author has implemented in scores of organizations. Two case studies on successful data quality projects and programs, respectively, are also presented.

Data quality assessment is an undeniably important component of data management practices as it provides the baseline against which improvements can be measured. However, due to the diversity in data quality dimensions (such as accuracy, completeness, and timeliness) and associated metrics, precise measurement of data quality costs and benefits is very challenging, leading to difficulties in preparing a business case for data quality initiatives. Towards this end, Markus Helfert and Ge Mouzhi provide foundation concepts on data quality cost and benefit measurement in the chapter “Cost and Value Management for Data Quality.”

The importance of sustainable practices for data quality management has led to significant study of data governance and organizational movements towards

maturity. In the chapter “On the Evolution of Data Governance in Firms: The Case of Johnson & Johnson Consumer Products North America,” Boris Otto outlines the principles of data governance and specifically discusses the evolution of data governance and ways to measure the changing organizational capabilities through a widely used maturity model.

# Data Quality Management Past, Present, and Future: Towards a Management System for Data

Thomas C. Redman

**Abstract** This chapter provides a prospective look at the “big research issues” in data quality. It is based on 25 years experience, most as a practitioner; early work with a terrific team of researchers and business people at Bell Labs and AT&T; constant reflection on the meanings and methods of quality, the strange and wondrous properties of data, the importance of data and data quality in markets and companies, and the underlying reasons that some enterprises make rapid progress and others fall flat; and interactions with most of the leading companies, practitioners, and researchers.

## 1 Introduction and Summary

From my 25 years experience in the field of data quality, I conclude that the most important research issues involve what I call the “management system for data,” loosely defined as the totality of effort to define and acquire data, ensure they are of high quality, store them, process and otherwise manipulate them, and put them to work to serve customers, manage the organization, and create new products and services. The conclusion stems from the following train of thought:

1. Many companies have made solid improvements.
2. Still, the vast majority has not. My review suggests four reasons:
  - (a) The business case for data quality is not very good.
  - (b) Social, political, and structural issues get in the way.
  - (c) Data have properties unlike other assets, but organizations have yet to take these into account in the data quality programs.
  - (d) It is not yet clear how organizations “make money” from their data assets.

---

T.C. Redman (✉)  
Navesink Consulting Group, Rumson, NJ, USA  
e-mail: [tomredman@dataqualitysolutions.com](mailto:tomredman@dataqualitysolutions.com)

3. Taken together, we need a comprehensive management system for data to fully address these root causes.

Digging deeper, many companies have made and sustained order-of-magnitude improvements. They've done so by focusing their efforts on "the most important needs of the most important customers," managing data quality at the points of data creation, and consistently applying relatively simple techniques to find and eliminate the root causes of error. No question that this has proven demanding work. But the benefits, in terms of lower costs, improved customer satisfaction, and better, more confident decision-making, have proven enormous. At these companies, the management system for data quality has proven more than adequate.

Still most enterprises, including both private section companies and government agencies, have not improved. Some are unaware of the importance of high-quality data. These are not of concern here. I am, however, concerned about the many that agree high-quality data are critical, but still have not improved.

I believe there are two major near-term reasons and two longer-term issues. First, some executives and leaders simply do not believe the "business case for data quality." And indeed, our business cases aren't very good.

Second, it is easy enough to spot the social, political, and structural issues that get in the way in any organization. For example, too many enterprises assign accountability for data quality to their Technology groups. In some respects, it is the natural choice. But Tech does not own the business processes that create data. So it is not well positioned to make needed process improvements. There are many such issues. Writ large, one can only conclude that the current management system in many companies and government agencies impedes data quality management and improvement.

Third, and longer-term, data have properties unlike other assets and these present both enormous opportunities and challenges for management. For example, data can be readily shared, across the enterprise and beyond, to an almost limitless degree.

More subtly, the most important (and least understood) is that each organization's data are uniquely its own. No other enterprise has, or can have, the same data. This is critical because it is the ways companies differ that offer opportunities for competitive differentiation. But few even think about their data in this way. Enterprises need better management systems so they can leverage some such properties and accommodate others.

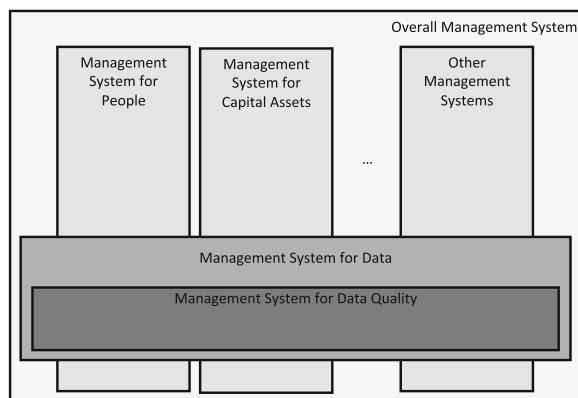
The fourth and final reason stems from the simple observation that, except in specialized areas,<sup>1</sup> markets are yet to reward those who provide high-quality data and punish those who do not. Further, we are only beginning to understand how enterprises "make money"<sup>2</sup> from their data. It's clear enough that the opportunities

---

<sup>1</sup>Specialized data providers serve many industries. Bloomberg, Morningstar, and Thomson-Reuters are household names in the financial services sector, for example. In these "pure data markets," data quality is indeed front and center.

<sup>2</sup>Quite obviously, government agencies and other nonprofits should not aim to "make money" from data. For them, "advance organizational mission" might be more appropriate. I've purposefully left

**Fig. 1** Management systems. Organizations have many management systems, most of which overlap with one another. Most pertinent to this chapter, the management system for data touches all other management systems, and the management system for data quality is an important component



are legion but, by and large, they are yet to crystallize. In parallel, most enterprises are still organized for the Industrial Age. The business models, strategies, types of people needed, organizational structures, and “data-driven cultures” all must be worked out.

To be clear, I’m calling for an overall “management system for data” that builds on successes so far, addresses these issues, and is fully integrated with and advance the enterprise’s other management systems (see Fig. 1).

## 1.1 *This Chapter*

As noted above, many enterprises have made significant data quality improvements. These successes provide a point of departure, so the next section of this chapter summarizes what those with the best data have done and why their efforts worked. The subsequent section examines why other organizations have failed. In other words, it expands on the issues noted above. The fourth section more clearly defines the research issues and describes further points of departure for the research. The final section argues that we have reached a point of crisis in data quality. So this work is extremely urgent.

## 2 Foundations and What Works

We’ve made solid progress in developing the technical and managerial underpinnings of data quality. And these foundations can be extremely effective. Organizations that apply “diligent but not heroic efforts” often improve data quality by one

---

“make money” in the body of text here, as I want to leave a hard edge on the point. Sooner or later, data must be recognized as equally important as capital and people (and maybe a few other) assets.

to two orders of magnitude. Figure 2 illustrates the sort of improvement I am talking about.

Table 1 summarizes how these leaders think about data quality and address the issues. I'll expand on the most salient points in the following section (for more details, see [26]).

## 2.1 *Data Quality Defined*

The notion of quality is inherently customer- (or user-) centric. A collection of data is of high quality, in the customer's eyes, if and only if it meets his, her, or its<sup>3</sup> needs. It is perhaps the simplest way of thinking about quality and it is certainly the most powerful, but it has profound implications. It means that data quality is inherently subjective. It means that different customers, even those whose needs are only subtly different, can rate a collection of data differently. It leads us to recognize that customers will increasingly demand data better and better tailored to meet their specific needs.

A more formal definition of quality embraces the concept of “fitness for use” [16]. In particular:

Data are of high-quality if they are fit for use in their uses (by customers) in operations, decision-making, and planning. They are fit for use when they are free of defects and possess the features needed to complete the operation, make the decision, or complete the plan.

Loosely speaking freedom from defects means the “data are right,” while possessing needed features means “having the right data.” These notions are made more objective via so-called dimensions of data quality, which I'll return to shortly.

Projecting the fitness-for-use concept into the future leads to the following “aspirational definition of data quality”:

Exactly the right data and information in exactly the right place at the right time and in the right format to complete an operation, serve a customer, make a decision, or set and execute strategy [26].

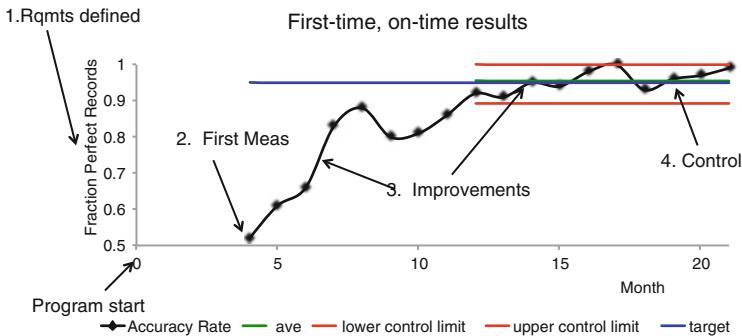
Both (customer-facing) definitions underscore both the enormous challenge and the enormous opportunity. Of course, the aspirational definition lays out an impossible mission. So those with the best data adopt the more practical, “day-in-day-out definition”<sup>4</sup>:

Meeting the most important needs of the most important customers.

---

<sup>3</sup>We explicitly recognize that a customer need not be a person. A computer program, an organization, and any other entity that uses data may qualify.

<sup>4</sup>The late Dr. William Barnard of the Juran Institute introduced me to this notion.



**Fig. 2** Typical data quality results for those applying “ten habits.” In this example, each error not made saves an average of \$500, amounting to millions quickly!

**Table 1** How those with the best data do it

Approach = How they think about quality:

Focus on *preventing* errors at their sources

Ten habits = How they do it:

1. Focus on the most important needs of the most important customers
2. Apply relentless attention to process
3. Manage all critical sources of data, including suppliers
4. Measure quality at the source and in business terms
5. Employ controls at all levels to halt simple errors and move forward
6. Develop a knack for continuous improvement
7. Set and achieve aggressive targets for improvement
8. Formalize management accountabilities for data
9. A broad, senior group leads the effort
10. Recognize that the hard issues are soft, and actively manage cultural change

## 2.2 Approach

By and large, today’s approach and techniques adapt quality techniques developed for the factory floor to data.<sup>5</sup> Most important is “approach.” Just as automobile manufacturers found they could produce higher quality cars at lower cost by “preventing future defects” rather than simply correcting them, so too data quality practitioners have achieved superior results by focusing their efforts at the points of data creation.

I believe this point is highly significant for future research. Especially in light of the ever-increasing rates of new data creation, and especially in light of the demands of “big data,” there is simply no substitute for “getting it right the first time.” And if

---

<sup>5</sup> And, to a lesser degree from good practice in data collection for scientific experimentation, though I know of no good reference to back up the assertion.

this is to happen, data creators must be held accountable for doing so (reference habit 8). Perhaps more than any other, this observation underscores the need for a properly aligned management system. If someone or some group other than data creators is accountable (or there is no accountability), efforts to improve data quality are ill-fated.

## 2.3 *A Management System for Data Quality*

The two most interesting moments in a datum’s lifetime are the moment it is created and the moment it is used. This is a bold, but extremely critical, assertion. Customer needs, at the moment of use, dictate what quality means and the moment of creation dictates whether customer needs will be met. The whole point of data quality management is to “connect these two moments” in effective and efficient ways. The ten habits help organizations do exactly that.

The day-in, day-out “work” is captured in habits 1, 4, 5, and 6 and could be rewritten as:

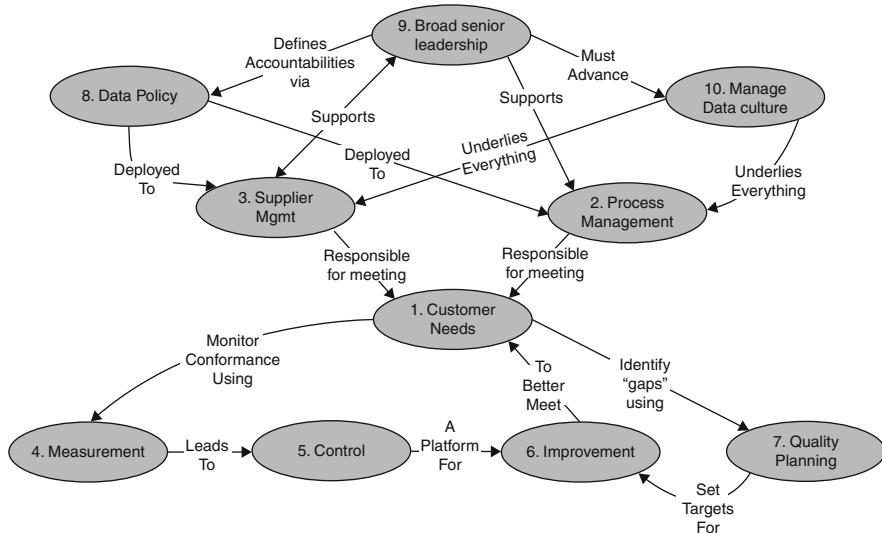
- (1) Understand who the customers are and their (most important) data requirements (as already discussed).
- (4) Measure against those needs [28].
- (6) Find and eliminate the root causes of error.
- (5) Implement controls to prevent simple errors from leaking through and to ensure gains achieved through process improvements are sustained.

Six-sigma enthusiasts will immediately recognize the parallels to the DMAIC (Define, Measure, Analyze, Improve, Control) cycle [25], perhaps the best-known method for uncovering and eliminating the root causes of errors.

At a higher level, process and supplier management have proven themselves unmatched at effecting the four working level habits. This point bears further explanation. While I’ve conducted no formal study, a “contributing factor” to (nearly) every data quality issue I’ve worked on is that data creators do not understand customer requirements. And no one should be surprised that data creators don’t meet needs they don’t understand. This is especially true when the creator works in one organizational “silo” and the customer in another.

One reason process and supplier management (habits 2 and 3) work so well is that they demand that and provide a formal means to connect creators and customers.

Habits 8, 9, and 10 bear on senior leadership. I’ve already noted at habit 8, putting responsibility where it belongs. Habits 9 and 10 stem from the simple observations that starting and sustaining a data quality program is hard work. Senior leadership (habit 9) is essential. And finally, habit 10 gets to the notion of “culture.” It recognizes that, if the actual work of data quality is challenging, changing mind-sets is well nigh impossible. But a data quality program cannot be sustained unless mind-sets change.



**Fig. 3** The ten habits reinforce on another. This figure adapted from Redman [27]

Leadership is so essential that habits 8–10 bear repeating, in a different fashion. Thus, people have to value both data and quality if an organization is to start, advance, and sustain a data quality program. Habits 8–10 summarize what leadership must actually do:

- (8) Clarify accountabilities for data. In particular, they hold data creators accountable for the data they create.
  - (9) Demanding leaders must put data quality near the tops of their lists of organizational priorities.
  - (10) Build and sustain a culture that values data and data quality.

Finally, it bears mention that all of this leads to definitive targets for improvement, set and pursued at all levels (habit 7).

Of course, in organizations with the best data, these tasks do not occur in isolation. They work together in a well-coordinated whole, or management system. Figure 3 presents a better graphic.

While other practitioners may express the above differently or have different emphases, I see no fundamental disagreement with these points among the leading practitioners, academics, and authors including, but not limited to, Aiken [36], Brackett [3], English [8], Eppler [9], Fisher [10], Hillard [13], Kushner and Villar [17], Lee [19], Loshin [22], McGilvray [23], Olson [24], Talburt [33], and Huang [14].

## 2.4 Data Defined

One area where there is considerable disagreement involves the definitions of data and information. Indeed, philosophers, computer scientists, statisticians, and others have debated the merits of different approaches for generations. While I'll not engage the debate here, I will present the approach to defining data that best reflects how data are created and used in organizations and so has proven itself remarkably suited to data quality.

In it, “data” consist of two components: a *data model* and *data values*. Data models are abstractions of the real world that define what the data are all about, including specifications of “entities” (things of the interest to the enterprise), important “attributes” (properties) of those things, and relationships among them. As an example, the reader is an entity. His or her employer is interested in its “employees” (an “entity class”), and attributes such as name, department, salary, and manager. Reports to is an example of a relationship between two entities.

The Taxing Authority is also interested in you (the reader), as a “taxpayer.” It is interested in some of the attributes that interest the employer, but many others, such as interest income, that do not. The reader is, quite obviously, the exact same person, but each organization has distinct needs and interests, so their data models are different.

On its own, a data model is much like a blank meeting calendar. There is a structure, but no content. Data values complete the picture. They are assigned to attributes for specific entities. Thus, a single datum takes the form: <entity, attribute, value>. As a specific example, consider <John Doe, Salary, \$35 K>. Here, John Doe is a specific entity belonging to the class entity employees. Salary is the attribute of interest, and \$35 K is the value assigned to John Doe for the attribute Salary. “Data” are any collection of datum of this form.

One last point on the definition of data: Clearly, data, defined this way, are abstract. We don't actually see or touch them. What we actually see when we work with data are data records presented in an almost unlimited number of forms: paper, computer applications, tables, charts, etc. The practical importance is that the same data can be presented in many different ways, to suit different customer groups.

It stands to reason that if data are to be of high quality, then its constituent parts (model, value, record) must be of high quality. And, in most enterprises, the constituent parts are created by separate processes:

- Data models are created by a data modeling process (they may also be obtained with enterprise systems or computer applications).
- Data values are created by ongoing business processes (these two may be obtained from outside sources).
- Data records/presentations are created in system and application development, processes that produce management reports, etc.

**Table 2** Dimensions of data quality

<b>Conceptual view/associated metadata</b>	<b>Data values</b>
Appropriate use	Accuracy
Areas covered	Completeness
Attribute granularity	Consistency
Clear definition	Timeliness
Comprehensiveness	
Essentialness	<b><i>Presentation quality</i></b>
Flexibility	Appropriateness
Homogeneity	Ease of interpretation
Identifiability	Formats
Naturalness	Format precision
Obtainability	Flexibility
Precision of domains	Handling of null values
Relevancy	Language
Robustness	Portability
Semantic consistency	Representation Consistency
Sources	Use of storage
Structural consistency	

Said differently, defining data in this way directs one to the processes that create their constituents (or the external suppliers of same). These are the processes (and suppliers) that must be managed and improved [20].<sup>6</sup>

## 2.5 Dimensions of Data Quality

I've noted repeatedly the importance of connecting data customers and data creators. One difficulty is that customers often speak in vague, confused, task-specific terms, while data creators need hard, tangible specifications on which to define and improve their processes. "Dimensions of data quality" help fill that gap, and the approach to defining data noted above provides a solid point of departure. Thus, one defines dimensions associated with the data model, with data values, and the data recording/presentation. Table 2 presents a list that was developed by the data quality team at Bell Labs in the early 90s [11, 21].

---

<sup>6</sup>A (perhaps) interesting historical note: In early 90s, the team I worked on at Bell Labs struggled to come up with a good definition of data, for quality purposes, and define dimensions of data quality. We finally came up with definitions we found acceptable. And then, we realized we had completely missed the point. Our approach treated data as "static," in a database. But static data are stunningly uninteresting. Data are interesting when they are created, moved about, morphed to suit individual needs, put to work, and combined with other data. We wrote [20] as a result and I personally think it is this team's most important paper. At the same time, the conclusion, once stated, is obvious!

Over the years, others have approached dimensions of data quality from different perspectives, and the best-known stems from early work at MIT [35]. I find that each organization likes to narrow the list and define its most important dimensions in its own way. Both Table 2 and the Strong/Wang work [35] provide solid starting points.

### 3 Why Aren't All Data of High Quality?

While we've made solid progress, we have a very long way to go. Even the best collections of data are far from perfect. Indeed, I know of no data collection that could credibly claim to be at a six-sigma level (3.4 defects per million data elements), the standard in simple manufacturing operations.

Worse, as noted, far too many organizations, even when they are aware of the proper approach to data quality and techniques to improve, are still saddled with poor data. They bear enormous costs and cannot leverage the value their data offer as a result.

I believe there are suggested four broad and possibly overlapping “root causes”<sup>7</sup>:

1. Some managers, executives, departments, and entire enterprises do not believe the rationale, i.e., that the benefits of poor quality dramatically outweigh the costs.
2. A combination of political, social, and structural issues makes it too difficult to define, grow, and sustain the data quality program.
3. Our data quality programs do not yet begun to fully embrace the “properties of data as a business asset,” which both make data so valuable and troublesome.
4. Immature data markets.

Let's consider each in turn.

#### 3.1 *Rationale Not Believed*

Frankly, I think one of the things we (those of us concerned about data quality) do really poorly is develop good business cases and deliver them in powerful ways. The current basic process involves:

1. Estimate the current “Cost of Poor Data Quality” (COPDQ). See Table 3 below for a list of areas where such costs occur.
2. Estimate on how much COPDQ can be lowered.
3. Estimate the cost of the effort to reduce COPDQ.
4. Assemble these into a business case.

---

<sup>7</sup>I've put “root causes” in quotes because a proper root cause analysis is considerably more disciplined than that conducted here.

**Table 3** List of increased “costs” due to poor data quality

<i>Operations</i>
Higher operating costs (non-value-added work)
Lower customer satisfaction
Lower employee morale
<i>Tactics/decision-making</i>
Lost sales
Lower trust between departments
Poorer and/or delayed decisions
Increased risk to acceptance of new technology
More difficult to manage overall risk
<i>Strategy</i>
More difficult to set and execute strategy
Fewer options to “put data to work”
Harder to align the organization
Distracts management attention
Threat to competitive position

The practical reality is that the only cost that can be estimated with any degree of precision is the increased costs in operations. While our lore suggests that total costs of poor quality far exceed these costs, we simply do not have reliable ways to estimate costs associated with angered customers, poor decisions, inability to manage risks, and so forth. And the effort to craft a business case never really gets off the ground.

To be fair, these difficulties predate data quality: Those promoting quality in manufacturing experienced many of the same problems we do.

### 3.2 Political, Social, and Structural Impediments

Interestingly, I find that almost everyone agrees that addressing data quality at the points of data creation is the only sensible way to approach the problem. And few people have any issues with the general thrusts of the ten habits. Still, a variety of political, social, and structural issues impede their enterprises’ efforts to improve data quality.

Quite naturally, data quality practitioners spend enormous amounts of time trying to understand and address the issues. Personally, I find the expression “all politics is local” enormously insightful. The people, and the details, differ in each enterprise, department, and work group. That said, I find six extremely common issues, as summarized in Table 4.<sup>8</sup>

---

<sup>8</sup>Several comments here. First, these are by no means the only issues. See Chapter 7 of *Data Driven* [26] for a fuller explanation of these and many others. Second, I am not the only person to have observed such issues. See Silverman [32] and Thomas [34] for other perspectives. Third, and most importantly, I have no formal training as a social scientist. It would be enormously helpful

**Table 4** Common social, political, and organizational impediments to data quality

Generic issue	Example implication
Brutal politics of data ownership/sharing	It is extremely difficult to improve data quality above the department level
Lack of accepted frameworks for privacy	Unclear who are legitimate customers for what data
Misalignment of dataflow and management	Accountability for data quality tends to drift from the data creator to the customer
Commingling management of data and technology	Tech departments are neither customers for nor creators of little much important data. There is little they can do to improve quality
Difficulties in defining and implementing standards	Few data standards make even the simplest communications across departmental lines more complicated
Misplaced accountability for data quality	Like Tech, there is little others can do to improve data quality at the points of data creation

While a full discussion is beyond scope, I do wish to illustrate how subtle, intertwined, and harmful they can be. As noted in the Introduction, many enterprises assign leadership of their data programs to their Tech groups.<sup>9</sup> In some cases, the rational is as simple as “Tech stores the data and moves it around a lot. They’re the natural people to lead the data quality program.”

This puts Tech in an uncomfortable position. On the one hand, Tech contributes to the data quality effort in many ways: It may lead data modeling efforts, it designs interfaces through which data are entered, it moves data around, it implements automated controls, and so forth. On the other hand, Tech is neither an important data creator nor customer. Most critically, Tech doesn’t own the business processes that create data and so can’t address issues at the points of creation. Nor is Tech well positioned to build communications channels between customers and data creators.

Instead, it (Tech) has no real option but to implement elaborate programs to find and fix errors. In doing so, it has, subtly perhaps, accepted responsibility for the quality. This is in direct opposition to the imperative that data be created correctly the first time!

They are other ways that responsibility for data quality can be mis-assigned. Data customers (users) are not blind to the ravages of bad data. So they set up hidden steps to “find and fix the bad data” inside their processes. In doing so, they have directly accepted responsibility for the quality of data they use, again in direct opposition to the imperative that data be created correctly the first time!

---

if sociologists, anthropologists, political scientists, and others brought more sophisticated tools to bear in helping understand these issues.

<sup>9</sup>Variously, Tech groups may be called Information Technology, the Chief Information Office, Information Management, Management Information Systems, etc.

Next, consider the overlapping issues associated with data sharing and personal/departmental power. My read of history is that the powerful have always tried to keep data for themselves, and with good reason. I once conducted a simple thought experiment using *The 48 Laws of Power* [12]. In the experiment, I imagined that I, as a middle manager, had just learned something that may be of interest to my colleague in the next office. I then read through each of the 48 laws, asking myself “Should I tell him or her?”

I see no applicability to 24 laws. But 23 are clear: “No, under no circumstances, go tell him or her.” The one remaining law advises, “It’s okay to go tell him or her. But get something in return that is even better.”

Of course, not everyone craves power. But the advice to those who do is clear. And the caution to others is just as clear!

Further, data sharing involves more than just allowing access to the data. It takes time and effort to explain what’s included, what’s not, the strengths and weaknesses, and the nuances in one’s data. And, in the vast majority of enterprises, there is simply no reward for doing so. Thus, lack of data sharing is not just a political issue. It is a structural one as well.

Now consider the misalignment of data flow and management. Data create value as they proceed “horizontally,” across enterprises. Yet, most enterprises are organized into functional silos, marketing, sales, operations, distribution, finance, etc. And most day-in, day-out management is “vertical,” up and down these silos. The net result is that day-in, day-out management and the use of data to create value are misaligned (even orthogonal to one and other).

Silos also make it more difficult to establish needed communications channels between data creators and customers (never mind more proactive data sharing!).

Finally, data standards: I’ll use a “common definition of ‘customer’” across a retail bank with multiple silos (savings accounts, car loans, mortgages, investor services) to illustrate the issue. As one would expect, the various silos developed their definitions of “customer” in ways best suited to their business lines. Thus, a “customer” might be variously thought of a “saver,” “homeowner,” and “investor.” Further complicating matters, any of these might be an individual, a couple, or a household. These different formulations make it difficult to determine who is who, across the silos. On one end of the spectrum, even a mundane question like “how many customers do we have?” is enormously complex. On the other end of the spectrum, business models based on a total view of the customer are difficult to execute.

Data standards are the obvious answer. But they have proven remarkably difficult in practice. Indeed, even the simplest proposed standard seems to engender opposition from all quarters. With good reason. Individual silos advance their business language, and hence their data models, to suit their business interests. Changing the data model threatens their business interests, for the uncertain returns of a common definition. So people are right to oppose data standards!

Politics aside, one can only conclude that deep structural issues make it more difficult to advance data quality. I sometimes claim “today’s organizational structures

are singularly ill-suited for data.” Indeed, it appears to me that inappropriate organizational structures exacerbate political issues. They demand a fundamental rethink.

### 3.3 *Properties of Data*

So far, data quality has borrowed much from manufacturing quality. This has provided a terrific platform on which to build. Further, as Blan Godfrey observes, “quality management is essentially the application of the scientific method to the problems of industry.”<sup>10</sup> So the thinking and ten habits have deep and established roots.

But the analogies to manufacturing only go so far, for data have properties unlike manufactured product (or any other asset for that matter). Take digitization. Quite simply, data can be digitized, then reformatted, copied, and shared to an almost limitless degree at extremely low cost. No other asset, certainly neither people nor capital assets, has this property. It offers unprecedented opportunity—to align the organization, to lower cost, and to increase the value of products and services. Thus (in principle), any customer can look at the data in his, her, or its preferred format (think product bar codes).

But with unprecedented opportunity comes unprecedented challenge. How, for example, can an organization manage and control the update all copies after one department makes a change? In another vein, the property makes it easier for a thief to steal data. He or she need not actually “take the data,” only make a copy and take it.

We’re only beginning to understand these properties and their implications. In some cases, we’ve developed solutions that are good enough for current problems. One example is “intangibility” and its impact on measurement. Quality management (and all of science for that matter) depends on measurement. But data have no physical properties, such as length, viscosity, or impedance, that admit physical measurement. Said differently, there is no “accurometer.”<sup>11</sup>

We’ve gotten around this issue in several ways. Perhaps, the best-known employs so-called business rules, which constrain the domain of values a datum (or pair, etc.) may take on. If the data values are outside the domain of allowed values, it is counted as incorrect. Other techniques employ expert opinion, comparison of data to their real-world counterparts, and data tracking and have proven their worth. But all have limitations and are unlikely to provide all the measurement horsepower needed

---

<sup>10</sup>Dr. Godfrey is Dean, School of Textiles, at North Carolina State University. He made the comment repeatedly as Head of the Quality Theory and Methods Department at Bell Labs and as CEO of the Juran Institute in the 1980s and 1990s. I don’t recall ever seeing it in print nor can I confirm that he was first to make the observation.

<sup>11</sup>I believe this observation is due to Robert W. Pautke, Cincinnati, OH.

**Table 5** Properties of data and an example implication for data quality

<b>Property of data</b>	<b>Example implication for data quality</b>
Data multiplies	The sheer quantities of data growth strain abilities to manage them and underscore the need to “get it right the first time.”
Data are more complex than they appear	Much (a proper data model, correct values, suitable presentation, etc.) has to go right for high-quality data to result
Data are subtle and nuanced. They have become the organization’s lingua franca	The distinction between the “right” data and “almost right” data may be akin to the distinction between lightning and a lightning bug (after Twain)
Data create more value when they are “on the move”	Data sitting in a database are profoundly uninteresting. Data quality efforts should be focused on the moments of data creation and use throughout the lifetime of the data
Data are organic	Useful data create new needs, leading to new data. Existing data morph as they move to suit different needs
Data can be digitized	They can be shared or stolen
Data are the means by which organizations encode knowledge. They are meta-assets	So they can have very long lifetimes
Data are intangible	They have no physical properties, complicating measurement
Each organization’s data are uniquely its own	Organizations distinguish themselves through the data they have that no one else does (and through their use of that data). These data are of special importance to the data quality program

going forward. Further, we have no good methods to measure the quality of most metadata (data definitions, etc.).

Data have many, many such properties, presenting both opportunity and challenge. Table 5 presents a more complete list.<sup>12</sup>

Again, while a full discussion of these properties is beyond scope here, I do wish to expand on one more. As I noted earlier, my intuition tells me that the last property, which an organization’s data are uniquely its own, is most important. It is only logical that “knowing something the other guy doesn’t” can be a critical, perhaps even decisive, source of advantage. Indeed, finding and exploiting a source of sustainable source of competitive advantage appears to be the essence of long-term success in capitalistic society.<sup>13</sup> This concept is front and center in the Information Age, as the successes of Google and Facebook bear witness.

---

<sup>12</sup>To be clear, I have every expectation that this list is incomplete.

<sup>13</sup>Note here another reason not to share data!

Data offer at least two pathways to uniqueness:

- Through the way the organization chooses to model its business in data (i.e., the way it defines critical entities such as customer, product, and transaction) and
- Through its business transactions, which add to its store of data every day.

Of course, not all data are unique. But it stands to reason that those that are, merit special attention. Enterprises must, in my view, look for ways to create and leverage unique data.

### ***3.4 Data Markets (or Lack Thereof!)***

It seems to me that the most consistent drivers of quality management have been market forces. For examples:

- Demands for huge quantities of materiel to support the World War II efforts drove American manufacturers in the 1940s.
- The fear of economic collapse after WW II drove Japanese automobile and consumer electronics industries to compete based on quality.
- Today, for many manufactured products, high quality is a condition of entry.
- In the rather few “pure data markets” that exist today, demands for high quality are high and growing.

But there are relatively few “data markets.” Indeed, most data are simply not for sale. The result is that data quality programs are, by and large, deprived of the most powerful motivators of other quality programs.

The situation is a bit more complex, of course. First, data contribute to enterprise success in many ways, even when they are not sold. Further, the opportunities to “make money” from data are only growing. But “how to do so” is far from clear. As noted in the Introduction, business models, strategies, demands on people, and organizational structures and culture all need to be worked out.<sup>14</sup>

## **4 Research Directions**

In [26], I proposed that for its data to qualify as an asset, an enterprise must take three steps:

1. It must take care of them. “Taking care” is mostly about quality, though security qualifies as well.

---

<sup>14</sup>Some may argue that these points merely reflect our progress in economic development and thus do not constitute a root cause at all. They have a fair point.

2. It must put its data to work, usually to make money, but more generally to advance the enterprise's interests.
3. It must advance the management system for data, taking into account the above and data's special properties.

This structure suggests a structure for describing my view of the big research directions, as follows:

- Bringing information theory to bear in quality improvement
- Monetizing data
- The management system for data

Taken together, these areas cover the root causes discussed above, though not in a one-to-one fashion.

#### ***4.1 Technical Foundations: Bring Information Theory to Bear***

Claude Shannon, for all intents and purposes, invented information theory in a sequence of two papers, “A Mathematical Theory of Communication,” in 1948 [30,31]. Information theory provides both a structure for thinking about information (information is defined in terms of entropy or uncertainty)<sup>15</sup> and the mathematical bases for measuring the quantity of information, data compression, error-correcting codes, and on and on. Information theory lies at the very heart and soul of the Information Age.

The problems of communications and data quality are very different: In communications, the problem is to faithfully reproduce the message, not use it. Still, the underlying theory is so powerful that it could advance nearly every aspect of data quality.

Let me trace a line of thinking that starts with control. I've already noted that controls are essential to data quality management. Many data quality controls stem from constraints on the domain of allowed data values. For example, in Australia,

$\text{STATE} \in \{\text{New South Wales, North Territory, Queensland, South Australia, Tasmania, Victoria, Western Australia}\}$

Values for the attribute STATE outside this domain cannot be correct. Domains of allowed values become increasingly complex with more attributes. Further, the wider the domain, the greater entropy.<sup>16</sup> Good controls reduce entropy. Ideally, one would like to evaluate the overall entropy reduction provided by a set of controls against a theoretical maximum, improve that set of controls, and make trade-offs (is the reduction in entropy worth the effort?).

---

<sup>15</sup>Earlier, I noted that there was considerable debate on the definition of “information” in our field. I think definitions should be based on entropy and/or uncertainty.

<sup>16</sup>I want to be careful here. This statement is not strictly true, as entropy is a probabilistic measure.

**Table 6** Research topics: bringing information theory to bear

- 
1. Is information theory applicable to data quality? Said differently, does the fact that information theory is disinterested in the underlying meaning of a signal while meaning is critical to data quality matter?
  2. How does one link entropy to data modeling? If that is possible, how does one apply the concept to create better data models? How does one apply the concept to deriving the most important messages in unstructured data?
  3. How does one link entropy to control and data quality measurement?
  4. What are the implications of the management system?
- 

Alternatively, one may wish to use entropy to opine on the quality of a datum or collection of data.

Now link to data modeling: In the example above, since STATE (in Australia) can only take on seven values, it can be coded using three bits, less than one English letter. But is that best? Might more facilitate “in-process” controls (and thus minimize “fat-finger” errors in data entry), including error-correcting codes? The list of possibilities is endless.

Now link data modeling to unstructured data: As I defined them, data impose structure on the real world, and in doing so, reduce entropy. Conversely, finding meaning in unstructured data<sup>17</sup> involves reducing the entropy, in effect imposing structure. This suggests that algorithms, optimized to reduce entropy, might lead to clearer interpretations of unstructured data.

Finally, link control to measurement: Recall my earlier observation that data have no physical properties, complicating measurement. We’ve gotten around the problem in a variety of ways, often involving business rules.<sup>18</sup> Still, we don’t have a foundation for evaluating how good our measurements are, never mind developing better ones. Information theory may provide the needed foundation.

With this backdrop, Table 6 summarizes my research questions.

I know of two attempts to bring the power of information theory to bear in data quality. Robert Hilliard [13] uses the measures of entropy to compare the so-called Inmon and Kimball architectures for data warehouses. This thinking suggests that one might be able to use entropy reduction as a means of evaluating data models throughout the end-end modeling process. Said differently, a data model imposes structure on an uncertain world, in effect reducing uncertainty. Day-in, day-out business language also implies a structure on the real world, though one that is excessively noisy. A “good” conceptual data model captures the essence of business language while at the same time reducing noise. Information theory may provide a solid basis for quantifying and advancing this thinking.<sup>19</sup>

---

<sup>17</sup>The careful reader will object that “if data are structured,” then “unstructured data” is nonsensical. Unfortunately, those who coined the phrase appear not to have taken this into account.

<sup>18</sup>Even data tracking, in many ways, the most powerful measurement technique employs a form of business rules.

<sup>19</sup>Rob Hilliard and I have started a research project along these lines.

**Table 7** Ways to bring data to market

<b>Concept</b>	<b>Example</b>
<i>Content providers</i>	
Provide new content	Personal diet regimen
Repackage existing data	Morningstar
Informationalization	GPS, directions in autos
Unbundling	Investment advice
Exploit asymmetries	Hedge funds
Close asymmetries	Internet price services
<i>Facilitators</i>	
Own the identifiers	S&P (cusip)
Infomediation	Google
Privacy and security	Privacy advocates
Analytics, data mining, big data	Amazon
Training and education	Internet training
New marketplaces	eBay
Infrastructure technologies	New database technologies
Information appliances	iPod
Tools	Workflow, SAS products

John Talburt [33] also introduces Shannon entropy in this book on entity resolution. Interestingly, however, the models he introduces take on probabilistic and algebraic, not entropy-based, formulations.

## 4.2 Monetizing Data

Inside enterprises, “better data means better decisions.”<sup>20</sup> The phrase is directionally correct, though of course “better data” also means better strategies, better alignment, increased ability to manage and improve operations, and on and on. All of these contribute to the bottom line. So cultivating a “fact-based decision culture,” a “data-driven culture,” or the like is one great way to monetize data. We’ve a lot of work to figure out the full implications. (I include it on my research issues in the next section.)

My earlier observation that there are no markets for most data notwithstanding, data play many and stunning roles in markets.<sup>21</sup> Table 7 provides a list, updated from [26]. And while each has deep historical roots, the business models for each are nowhere near developed. Working these out is critical to the advance of the Information Age.

---

<sup>20</sup>This phrase closely mirrors the moniker of Data Blueprint, “better data for better decisions.”

<sup>21</sup>“Stunning” in the sense that these roles were so unexpected even a few years ago.

**Table 8** Monetizing data

- 
1. What are the business models for each of the ways of putting data to work in the marketplace?
  2. Are there other ways to put data to work?
  3. How should an enterprise decide which are the most important?
  4. Can one use employ “internal data markets” as a means to stimulate quality improvement?
  5. How does one assign economic value to data and data quality? What are the limits to valuation methods?
  6. What are the market requirements for data quality? What are the implications for data quality management?
  7. What are the implications for the management system for data?
- 

Table 8 summarizes my research questions.

I’m aware of three separate trains of thought that bear on the valuation question and so may provide points of departure. First, recent work by Brynjolfsson, Hitt, and Kim [4] suggests that the data-driven, those who use data more effectively in decision-making, are reaping productivity gains of 5–6 % given other factors.

Second, Cambridge University (see Borek, Parlakad, and Woodall [2]), is leading research in what it calls Total Information Risk Management. The idea is simple and appealing: Cast the data quality business case problem as one of assessing risk, borrow best-practice and accepted process from areas where risk assessment/management is more fully developed, and adapt them to the specifics of data quality.

The third potentially fruitful train of thought borrows from accounting. If, as many of us claim, “data are assets,” then one ought to be able to place an accounting value on them. Doug Laney [18] has done some initial thinking in this area. He explores what “makes it to the balance sheet” and explores six possible ways to assign a business value to data quality:

1. Intrinsic value
2. Business value
3. Loss value
4. Performance value
5. Economic value
6. Market value

### **4.3 Fundamental Rethink of the Management System for Data**

I’ve repeatedly opined that the issues and opportunities described herein call for a fundamental rethink of the management system for data.<sup>22</sup> After Roberts [29],

---

<sup>22</sup>Some may argue that one cannot complete a “fundamental rethink” in advance of a “fundamental think.” They have a point.

I include people, organization structure, management routine (largely governance), and culture as components of the term “management system.”

Of course, “a fundamental rethink the management system for data” is overly broad. Table 9 presents my current formulation of my research topics in this space.

There are several points of departure for research into these topics. First, what works now? Fig. 4 presents “the fundamental organization unit” (or building block) for data quality. It recognizes that data are either obtained from outside the organization or created inside, via the enterprise’s business processes. Thus, both supplier management and process management are essential. And, as noted earlier, these structures are ideal for organizing the day-in, day-out work of data quality. In particular, they are ideally positioned to charter improvement teams, each aimed at identifying and eliminating a specific root cause of error. Finally, the figure makes clear the need for senior leadership and technical support.

Figure 5 looks at a higher level and presents our “current best organization for data.” The concept is straightforward—scale Fig. 4 up by overlaying the structures needed to affect the ten habits across the entire enterprise. Thus, in the center of “current best organization for data” are process and supplier management structures.

A Chief Data Office, separate from the Technology group, defines the overall program and moves it along, and a Data Council (the higher and broader the better) oversees the work. For example, the Data Council ensures that the right processes and suppliers are identified and managed.

Like the ten habits, the “current best” is proving successful in that where components are diligently applied; better quality data almost always results. So Fig. 5 represents an important point of departure. But the “current best” suffers from the same lack of penetration as the ten habits and does not fully address my research questions.

A second point of departure is the management systems for other assets. As organizations have successfully managed people and capital assets for generations, the idea is to adapt what already works for these assets to data. For example, a typical (federated) management system for people features the following:

1. Most actual human resource management is done in the line, by line managers.
2. Every employee must contribute in a variety of ways, from the review process, to developing themselves through training, to understanding and following the policies and procedures of the company.
3. The human resources department has some “line responsibilities,” such as managing payroll and the benefits plan.
4. Most HR work involves setting policy and administering processes around those policies, while line managers are responsible for implementation. Performance review is a good example.
5. Dedicated staff. My informal studies yield 1–2 % of the total population, in enterprise or departmental HR jobs. Importantly, HR is an accepted “profession,” a point worthy of further discussion at some point.
6. An enterprise HR function has a very senior head that is involved in corporate level decision-making.

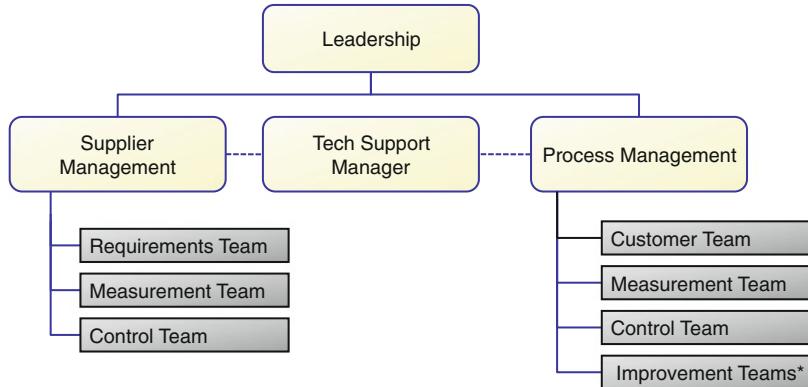
**Table 9** Research questions to develop better management systems for data

1. *Size of effort:* How many people are needed?
2. *People:*
  - What are the “basic,” “advanced,” and “world-class” data skill sets for knowledge workers?
  - What are the equivalent skill sets for managers?
  - How do existing employees advance their skill sets?
  - How do organizations manage and leverage new employees with more advanced skill sets?
3. *Organization structure:*
  - Where do they work?
  - What is the best (or even a better) overall organizational structure for data?
  - Does data demand entirely different management structures (i.e., process orientation)?
  - To whom does “data” report?
  - Do the properties of data and the new business models for monetizing data demand fundamentally different organization structures?
4. *Decision rights (governance):*
  - What are the (high-level) job descriptions?
  - And decision rights?
5. *Management accountabilities:* Since everyone “touches” data in some way and so can affect their quality, how should the management accountabilities be defined and deployed throughout?
6. *Connecting data customers and creators:* How should an organization ensure that data creators understand customers’ quality requirements?
7. *Interaction with other management systems:*
  - Are there special considerations for “management data” (used to run the organization)?
  - How does the management system for data interact with the other management systems? Ex: HR. How does an organization build data into the human resources management system?
8. *Data sharing:*
  - How should an organization promote and/or demand data sharing?
  - Under what terms and conditions?
  - Is a special organizational unit required?
9. *Special provisions for unique data:* How should an organization identify, acquire, nurture, protect, and utilize (monetize) data that are uniquely its own?
10. *Standard data:* How should an organization resolve the need for “standard data” to promote cross-departmental communication with the need of individuals for highly-nuanced data to complete their tasks?
11. *Metadata:* Are special methods for development and promulgation and use of metadata needed? If so, what are they?
12. *Privacy:* How should the organization think through its privacy obligations and/or what is just plain “smart business?”
13. *Culture:*
  - What are the key features of a “data-driven” culture?
  - How does one advance such a culture?

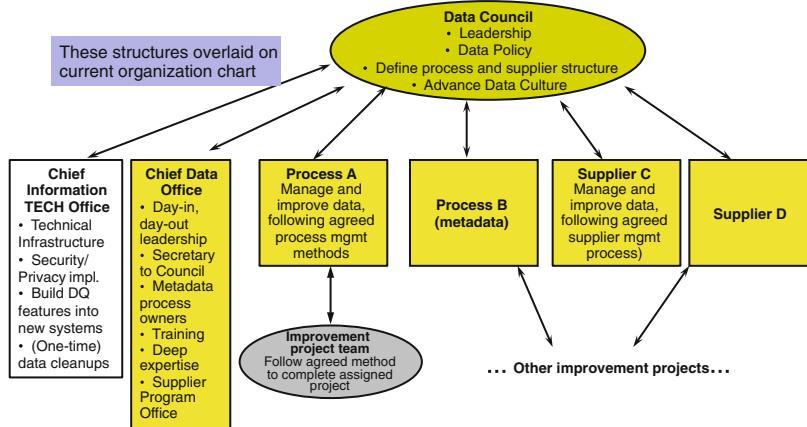
*What am I missing:* It seems extremely unlikely that my understandings of political/social issues and properties of data are comprehensive and/or explained in the most straightforward ways

---

*Note:* I reformulate these constantly. This is my list as of December 1, 2012



**Fig. 4** The fundamental organizational unit for data quality. \*Quality improvement facilitator is a permanent role, supporting a series of project teams, which disband when their projects complete.



**Fig. 5** Current "best" overall organization structure for data quality. This figure adapted from Redman [27]

A third point of departure is the development of data management (even if not called that) throughout history [6,7]. There is something age-old about my research questions, for an organization's most senior management has always concerned itself the flow of some data, to coordinate the army, to administer the Empire, and to allocate resources to Divisions. Indeed, management itself would not be possible without data flowing up and down. It may well be that the hierarchical form (and the resulting silos) was the optimal way to organize for such data flow. Even today, the hierarchical form offers at least one distinct advantage: People and departments have deep understanding of the data they use. So a better management system may require simple rearrangements to promote horizontal data flow.

In this respect, I'm particularly intrigued by the work of Stafford Beer [1], who applied cybernetics to organization. Of particular interest are the data flows needed by management to control and coordinate operations, create synergy, innovate, and promote organizational identity.

I'm also intrigued by the work of Eliot Jacques [15], who looks at organization through the lens of individual's abilities to process data. Those higher in the hierarchy can, in principle anyway, understand increasingly complex data, assimilate data from more and more even disparate sources, and project implications over longer and longer time horizons.

A final point of departure is social networks. The last 10 years have seen an explosion of social networking tools and organizations are grappling with the implications. I'm especially interested in the scientific community, an older and "slightly more disciplined" social network. From a data quality perspective, scientists do several things well:

- They understand the importance of "new data" to scientific discovery.
- They define and manage data collection processes with great care. They build controls into these processes.
- They carefully define terms (i.e., metadata).

Further, the scientific community is often better than others in sharing data. It may hold some clues for a better management system for data.

## 5 Final Remarks: Tremendous Urgency for This Work

To conclude, I wish to remark on the urgency of this work. At least since I've been involved, poor data have bedeviled (almost) every organization, increasing cost, angering customers, compromising decision-making, and slowing innovation. Worse, data quality issues contribute mightily to all of the important issues of our time, from the global war on terror (e.g., the case for the Iraq War), to trusted elections (e.g., the presidential election of 2000), to the roots of financial crisis, and on and on.<sup>23</sup>

But it seems to me that today, in 2013, there is even greater urgency. In a slightly different vein, technological capabilities to acquire data, store them, process them, reformat them, and deliver them to the farthest reaches have run far ahead of our abilities to manage and put the data to work. Nothing odd about that—technological progress often runs far ahead of management's (and/or society's) ability to utilize the technology. And new technologies cannot achieve their full potential until management catches up. Nicholas Carr [5] uses electricity and the electric grid to illustrate this point.

---

<sup>23</sup>See Chapter 3 of *Data Driven* [26] for a more complete discussion.

Finally, most of the world is struggling to regain its economic footings. I am, of course, biased. But I don't see how that happens in a broad-based, sustainable way unless more and more companies create new products and services. In other words, innovate. All things data, including big data, advanced analytics, and on and on, are front and center in the needed innovation.

Bad data stifle innovation, mostly in millions of small ways. They make it harder to see the deeper meanings in the data, they add risk throughout the process, and they bleed away resources that should be directed at innovation.

So, it is not hyperbole to claim that we are in data quality crisis. This data quality crisis may be even more severe than the quality crisis that hindered US economic growth in the 1980s. One of the most important lessons from quality in manufacturing is that "you can't automate your way out of a quality crisis." That takes management.

## References

1. Beer S (1979) *The heart of enterprise*. Wiley, New York
2. Borek A, Parlikad AL, Woodall P (2011) Towards a process for total information risk management. In: Proceedings of the 16th international conference on information quality, University of South Australia, Adelaide, 18–20 November 2011
3. Brackett MH (2000) *Data resource quality turning bad habits into good practice*. Addison-Wesley, Boston
4. Byrnjolfsson E, Hitt LM, Kin HH (2011) Strength in numbers: how does data-drive decision making affect firm performance? SSRN: <http://ssrn.com/abstract=1819486> or <http://dx.doi.org/10.2139/ssrn.1819486>
5. Carr N (2003) IT doesn't matter. *Harv Bus Rev* 81(5):41–49
6. Chandler AD (1977) *The visible hand the managerial revolution in American Business*. The Belknap Press, Cambridge
7. Chandler AD, Cortada JW (eds) (2000) *A nation transformed how information has shaped the United States from colonial times to present*. Oxford University Press, England
8. English LP (1999) *Improving data warehouse and business information quality*. Wiley, New York
9. Eppler MJ (2003) *Managing information quality*. Verlag, Berlin
10. Fisher T (2009) *The data asset: how smart companies govern their data for business success*. Wiley, Hoboken
11. Fox C, Levitin AV, Redman TC (1994) The notion of data and its quality dimensions. *Inf Process Manag* 30(1):9–19
12. Greene R, Elffers J (1998) *The 48 laws of power*. Viking, New York
13. Hillard R (2010) *Information-driven business: how to manage data and information for maximum advantage*. Wiley, Hoboken
14. Huang KT, Lee YW, Wang RY (1999) *Quality information and knowledge*. Prentice-Hall, Upper Saddle River
15. Jacques E (1988) *Requisite organization*. Cason Hall & Company, Arlington
16. Juran JM, Godfrey AM (1999) *Juran's quality handbook*, 5th edn, McGraw-Hill, New York
17. Kushner T, Villar M (2009) *Managing your business data: from chaos to confidence*. Racom Communications, Chicago
18. Laney D (2011) Infonomics: the economics of information and principles of information asset management. In: Proceedings of 5th MIT information quality industry symposium, Cambridge Massachusetts, 13–15 July 2011

19. Lee YW, Pipino LL, Funk JD, Wang RY, (2006) *Journey to data quality*. MIT Press, Cambridge
20. Levitin AV, Redman TC (1993) A model of data (life) cycles with applications to quality. *Inf Softw Technol* 35(4):217–224
21. Levitin AV, Redman TC (1995) Quality dimensions of a conceptual view. *Inf Process Manag* 31(1):81–88
22. Loshin D (2011) *The practitioner's guide to data quality improvement*. Elsevier, Amsterdam
23. McGilvray D (2008) *Executing data quality projects ten steps to trusted data*. Morgan Kaufmann, Amsterdam
24. Olson JE (2009) Data quality the accuracy dimension. Morgan Kaufmann, Amsterdam
25. Pyzdek T, Keller P (2009) *The six-sigma handbook*. 3rd edn. McGraw-Hill, New York
26. Redman TC (2008) *Data driven: profiting from your most important business asset*. Harv Bus Press, Boston
27. Redman TC (2001) *Data quality: the field guide*. Digital Press, Boston
28. Redman TC (2004) Measuring data accuracy: a framework and review. *Stud Commun Sci* 4(2):53–58.
29. Roberts DJ (2004) *The modern firm*. Oxford University Press, Oxford
30. Shannon CE (1948) A mathematical theory of communication. *Bell Syst Tech J* 27:379–423
31. Shannon CE (1948) A mathematical theory of communication. *Bell Syst Tech J* 27:623–656
32. Silverman L (2006) *Wake me when the data is over: how organizations use stories to drive results*. Jossey-Bass, San Francisco
33. Talbut JR (2011) *Entity resolution and information quality*. Morgan Kaufmann, Amsterdam
34. Thomas G (2006) *Alpha males and data disasters: the case for data governance*. Brass Cannon, Orlando
35. Wang RY, Strong DM (1996) Beyond accuracy: what data quality means to data consumers, *J Manag Inf Syst* 12(4):5–33
36. Yoon Y, Aiken P, Guimaraes T, (2000) Managing organizational data resources: quality dimensions. *Inf Resour Manag J* 13(3):5–13

# Data Quality Projects and Programs

Danette McGilvray

**Abstract** Projects and programs are two fundamental ways of putting data quality into practice. A data quality (DQ) project includes a plan of work with clear beginning and end points and specific deliverables and uses data quality activities, methods, tools, and techniques to address a particular business issue. A data quality program, on the other hand, often spearheaded by an initial project, ensures that data quality continues to be put into practice over the long term. This chapter focuses on the components necessary for successful data quality projects and programs and introduces various frameworks to illustrate these components, including the Ten Steps to Quality Data and Trusted Information™ methodology (Ten Steps™). A discussion of two companies—one housing a mature data quality program, the other a more recent “DQ start-up” initiative—shows two examples of how data quality components and frameworks were applied to meet their organizations’ specific needs, environments, and cultures. Readers should come away from the chapter understanding the foundation behind the execution of data quality projects, the development of data quality programs, and generate ideas for incorporating data quality work into their own organization.

## 1 Introduction

This chapter is written as if talking to the people responsible for designing, implementing, and leading a formalized data quality program within their organization.<sup>1</sup> The person who heads up this initiative is called the “data quality program manager.”

---

<sup>1</sup>I may use the words business or company when referring to an organization. Realize that everything said here applies to any type of organization—for profits, nonprofits, government, education, and healthcare because all depend on information to succeed.

D. McGilvray (✉)  
Granite Falls Consulting, Inc., 39899 Balentine Drive, Suite 215, Newark, CA 94560, USA  
e-mail: [danette@gfalls.com](mailto:danette@gfalls.com)

Those who participate in the program are collectively the “data quality team” whether they are an actual team as defined in their human resources structure or a virtual team who has responsibilities to the DQ program manager but no direct reporting relationship. Program or project “sponsors” can also learn about the data quality work for which they may have ultimate accountability. “Stakeholders” whose other responsibilities can be helped with or impacted by data quality work can also find this chapter of interest. Finally, researchers and students of data quality will find the practical application of data quality theory in this chapter. We will get more into organizational issues, roles, and responsibilities later, but for now that sets the stage for the audience of this chapter.

Any of the DQ team members or the manager can be referred to as an information quality professional. This is a person who “holds any of a wide range of positions in their organizations, as individual contributors or as managers. They conduct, lead, or champion information quality projects. They work in any of the functions or disciplines within their organization or are part of a specialized information quality team; yet all perform information quality activities as part of their job responsibilities. This information quality work is either part-time within a broader organizational role, or full time” [1]. Please note that data quality and information quality are used interchangeably.

Years ago I was introduced to the idea of being an “intrapreneur”—a person who acts like an entrepreneur (initiates a new business enterprise) but does it within a corporate structure. The data quality program manager is an intrapreneur because she is creating a new business which is caring for one of an organization’s most important assets — data. To that end, learning how to apply management practices for being in business to the data quality program will enhance your chances of success.

The data quality team is part of the “start-up” company and will need all the enthusiasm and dedication that any new organization requires to get its feet on the ground. Every manager needs to know how to organize people, money, and other resources to meet his or her goals.

Every DQ team member needs to bring his or her own professionalism, expertise, and experience to ensure that the business has the quality of data needed to meet its objectives and goals, tackle issues, implement its strategies, and take advantage of opportunities. Lest you think the DQ team is the only one in charge of data quality, let me point out that one of the DQ team’s tasks is to identify the many people who affect the information throughout its life cycle. The DQ team then ensures those people are properly trained and motivated to take action on whatever their responsibilities are as it relates to data quality. Organizations must provide for the “care and feeding” of their data and information [7]. Understanding the difference and the relationship between data quality programs and projects is an essential element of that responsibility.

With the unprecedented increase in content spurred on by a variety of news and social media outlets, executives have identified multiple concerns. First and foremost in their minds is the quality and accuracy of data [4].

## 2 Starting Point: Program or Project?

Let's begin with two definitions and answer the question: What is a data quality program vs. a data quality project? A *program* is an ongoing initiative whereas a *project* is a “temporary endeavor undertaken to create a unique product, service, or result” [6]. A *project* has a plan of work with a beginning, a middle, an end, and specific deliverables. A *data quality program* is part of the management system for data and information that addresses business needs by providing services that will ensure ongoing information quality. A *data quality project* is a project that focuses specifically on a data quality problem or incorporates data quality activities into other projects, methodologies, or ongoing operations in order to help solve a business need.

Sustaining data quality within any organization requires both the ability to execute data quality projects and the ability to have a foundation that provides the support, tools, training, and resources (people, money, expertise) for ongoing data quality needs once a project is complete. Without the program, data quality work is at risk of not continuing; without the data quality projects, data quality will not be actually implemented within the company. Said in a more positive way—with a program, data quality work has a high chance of being sustained; with the data quality projects, data quality has a good chance of actually being created, improved, and managed within the company.

As you tackle data quality in your company you will see that the program and project aspects are actually intertwined. What is the starting point? Often motivated and informed individuals see the need for a focus on data quality. On their own they attend conferences, research industry best practices, take courses, and educate themselves on data quality. They start a data quality project assessing the quality of a dataset of interest to the area where they work. This might be key tables in a data warehouse, master data that forms a foundation for important business processes, or poor transactional data that is causing problems in carrying out day-to-day responsibilities. From there they see that managing data quality in their organization is more than one project or a series of projects. They realize it also requires a foundation to provide expertise, tools, services, and training to sustain data quality at the company. At this point they put effort toward building a data quality program, the ongoing initiative that will help take what they learned from their first data quality project and ensure that experience can be used in other areas of the company.

Even if your starting point is to build a foundational program, one of your first activities will be to carry out a pilot project. For this reason, let's discuss data quality projects first and a structured approach to executing data quality projects.

### 3 Data Quality Projects

For the purpose of this discussion, I use the word project in a very broad sense meaning any significant effort that makes use of data quality concepts, methods, tools, and techniques. A project is a task or planned set of work that requires time, effort, and planning to complete. It is the means by which you take action on solving a specific challenge. Your data quality project may be focused on:

- Addressing a specific data quality problem such as assessing the quality of a dataset of interest to the area where you work. As mentioned earlier, this might be key tables in a data warehouse or master data that forms a foundation for important business processes.
- Operational data or processes where you are responsible for data quality or the work you do impacts data quality such as receiving and loading data from external vendors into your production system on a regular basis.
- Incorporating specific data quality activities into other projects using other methodologies. For example, any data integration or migration project such as building a data warehouse for business intelligence and analytics use or implementing an ERP (Enterprise Resource Planning) system.
- Integrating data quality activities into your company's standard SDLC (software/solution development life cycle).

In each of these examples, applying a systematic approach for addressing your data quality challenges through projects will save your company much time and money. You can avoid the risks and costs of “reinventing the wheel” and instead use the specific knowledge and expertise of your company, environment, and culture to develop highly effective solutions related to data quality. You will have many issues related to data quality over time and having these skills will ensure the ability to anticipate and address many situations effectively when they arise.

### 4 The Ten Steps<sup>TM</sup> Methodology

Let me introduce you to one such systematic approach—Ten Steps to Quality Data and Trusted Information<sup>TM</sup> (Ten Steps<sup>TM</sup>). The Ten Steps methodology consists of a framework, concepts, and processes for improving, creating, and managing information and data quality. As an information quality professional, you need to understand the concepts underlying data quality so you can effectively apply a structured process such as the Ten Steps to the many situations where data quality can help your organization. To learn more about the Ten Steps methodology, see [2].

## 4.1 The Framework for Information Quality and Other Key Concepts

To apply the Ten Steps effectively, it is first necessary to understand some key concepts about information and data quality. The Framework for Information Quality (FIQ) shows the components necessary for information quality (see Fig. 1). Understanding concepts such as the information life cycle, data quality dimensions, business impact techniques, data categories, data specifications (your standards, data architecture and models, business rules, metadata and reference data; anything that gives your data structure and meaning), and data governance and stewardship are also essential. The concepts and FIQ provide a background and frame of reference to help us make sense of the world from a data quality point of view.

The FIQ is a tool for:

- **Diagnosis.** The FIQ helps us understand an existing complex environment. By using the components as a checklist, we can see what is happening in each area, assess our practices, and determine if the components necessary for information quality are being addressed or not.
- **Planning.** The diagnosis provides input to our planning and helps us determine where to invest time, money, and resources by showing us where breakdowns are occurring. These are high priority for focusing our efforts. We can also see where we are doing things well, which is equally important because it means we don't need to put effort in those areas. Sometimes people underestimate the importance of knowing what NOT to pay attention to.
- **Design.** We can use the framework to design new business processes or update existing processes. Do we know what is happening with the 4 key components through the information life cycle? Did we take location and time into account when we designed the process? Have we considered the broad-impact components of risk—RRISCC<sup>2</sup>? Do we understand the business need that we are fulfilling? Does what we have designed fit into our culture and environment in a way that it will actually be used?
- **Communication.** For some audiences seeing the actual framework is helpful. For others, being able to explain the components and their relationships is enough for them to understand that it takes effort and coordination of many parts to ensure high-quality information.

To summarize, the framework allows us to organize our thinking in a way so we can make good decisions in order to take effective action as it relates to our information

---

<sup>2</sup> See Fig. 1, RRISCC refers to broad-impact components, which are additional factors that affect information quality (Requirements and Constraints, Responsibility, Improvement and Prevention, Structure and Meaning, Communication, Change). You can lower your risk of poor data quality by ensuring the components have been appropriately addressed. If they are not addressed you increase the risk of having poor quality data.

Business Goals / Strategy / Issues / Opportunities (Why)					
	<u>Plan</u>	<u>Obtain</u>	<u>Store and Share</u>	<u>Maintain</u>	<u>Apply</u>
Data (What)					
Processes (How)					
People/Orgs (Who)					
Technology (How)					
Location (Where) and Time (When and How Long)					
<u>Requirements and Constraints:</u> Business, Technology, Legal, Contractual, Industry, Internal Policies, Privacy, Security, Compliance, Regulatory					
<u>Responsibility:</u> Accountability, Authority, Governance, Stewardship, Ownership, Motivation, Reward					
<u>Improvement and Prevention:</u> Root Cause, Continuous Improvement, Monitor, Metrics, Targets					
<u>Structure and Meaning:</u> Definitions, Context, Relationships, Standards, Rules, Architecture, Models, Metadata, Reference Data, Semantics, Taxonomies, Ontologies, Hierarchies					
<u>Communication:</u> Awareness, Out-Reach, Education, Training, Documentation					
<u>Change:</u> Management of Change and Associated Impact, Organizational Change Management, Change Control					
<b>Culture and Environment</b>					

Source: Copyright © 2005–2008 Danette McGilvray, Granite Falls Consulting, Inc.  
 Excerpted from *Executing Data Quality Projects: Ten Steps to Quality Data and Trusted Information™* by Danette McGilvray, published by Morgan Kaufmann Publishers. Copyright © 2008 Elsevier Inc. All rights reserved.

**Fig. 1** The Framework for Information Quality (FIQ)

quality. The Ten Steps process, which we will discuss next, is how we implement those concepts.

## 4.2 *The Ten Steps Process*

The Ten Steps process is the approach for creating, assessing, improving, and managing information and data quality. It is how you put the concepts into action to solve your business needs or issues, address goals and strategies, or take advantage of opportunities (see Fig. 2 the Ten Steps process).

The Ten Steps were designed so you select the applicable steps to address your particular needs. For any given project and situation, you will use different combinations of steps and go to varying levels of detail within the chosen steps. The following are typical circumstances where you can make use of the Ten Steps approach when you need to:

- Establish a business case for data quality in order to gain support for your program and projects
- Establish a data quality baseline to understand the current state of your data quality and set a reference against which to measure progress
- Determine root causes of data quality problems so you are treating the underlying causes and not just the symptoms
- Implement improvements that will prevent future data quality problems and correct existing errors
- Implement controls such as ongoing monitoring and metrics
- Address data quality as part of your daily responsibilities related to ongoing operations
- Integrate data quality activities into other projects and methodologies

Figure 3 highlights which of the Ten Steps would most likely be used in each of the approaches described above.

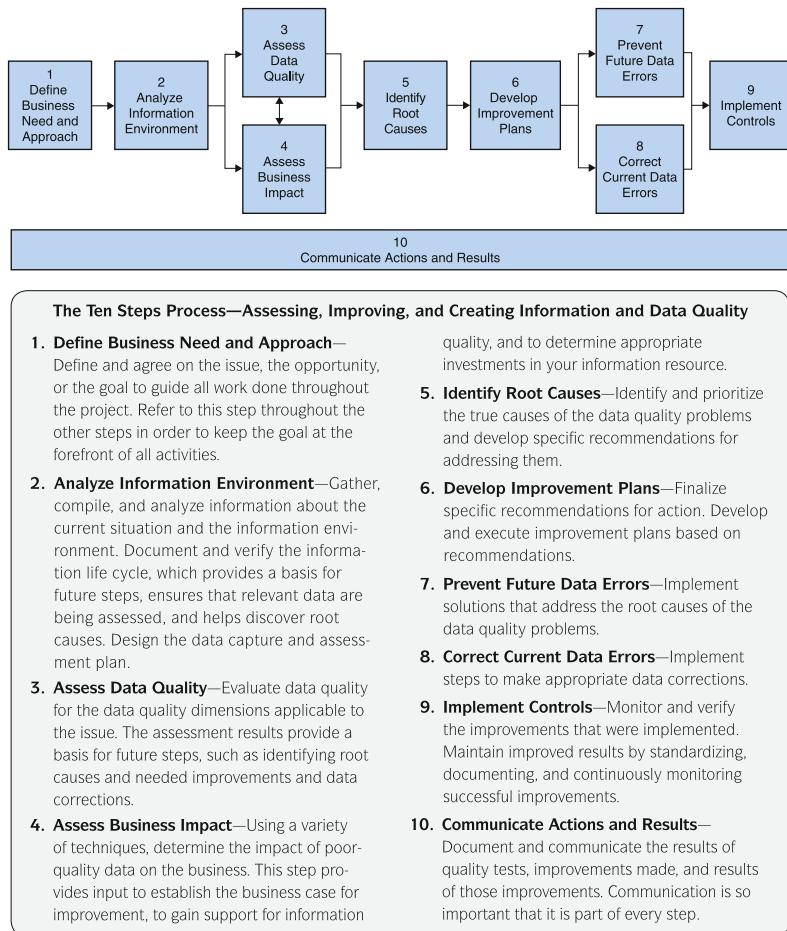
The Ten Steps process includes instructions, techniques, examples, and templates and provides enough direction so you can understand your options. It is up to you to decide which steps are relevant to your situation. Also an important factor to apply the steps successfully lies in determining the right level of detail needed for each step chosen.

We just discussed the Framework for Information Quality which showed the components necessary for data quality. The Ten Steps process is how those concepts and components are accounted for and applied through projects to real situations and challenges in your organization. Let's talk next about a Data Quality *Program* Framework.

## Overview of The Ten Steps Process

The Ten Steps process is the approach for assessing, improving, and creating information

and data quality. The steps are shown in the figure and described in the box.

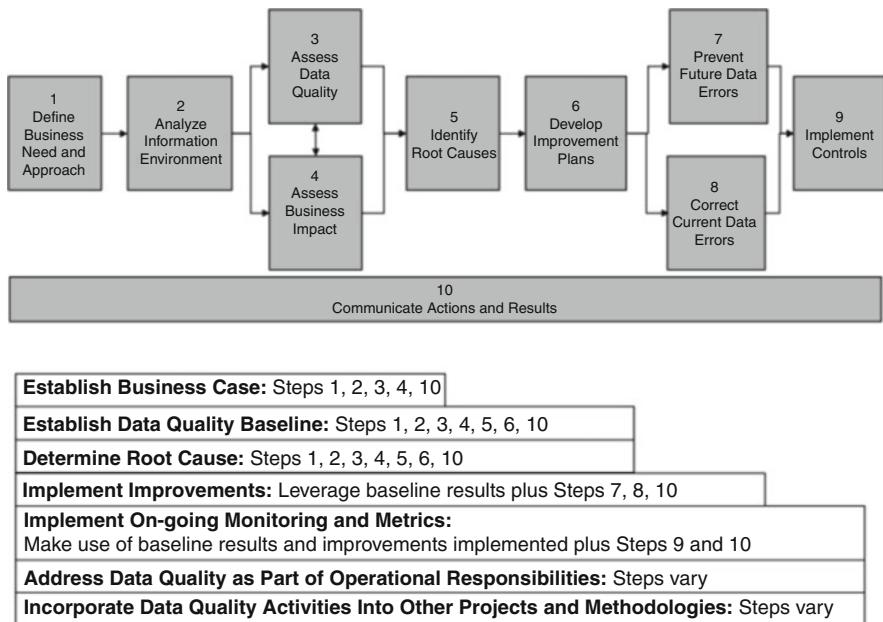


Source: Copyright © 2005–2008 Danette McGilvray, Granite Falls Consulting, Inc. Excerpted from *Executing Data Quality Projects: Ten Steps to Quality Data and Trusted Information™* by Danette McGilvray; published by Morgan Kaufmann Publishers. Copyright © 2008 Elsevier Inc. All rights reserved.

**Fig. 2** The Ten Steps process

## 5 Data Quality Programs

As already mentioned, it is not unusual that once a data quality team has done their first project or two they can see that without an ongoing program of some kind, data quality will be difficult to sustain. This is because once a project is

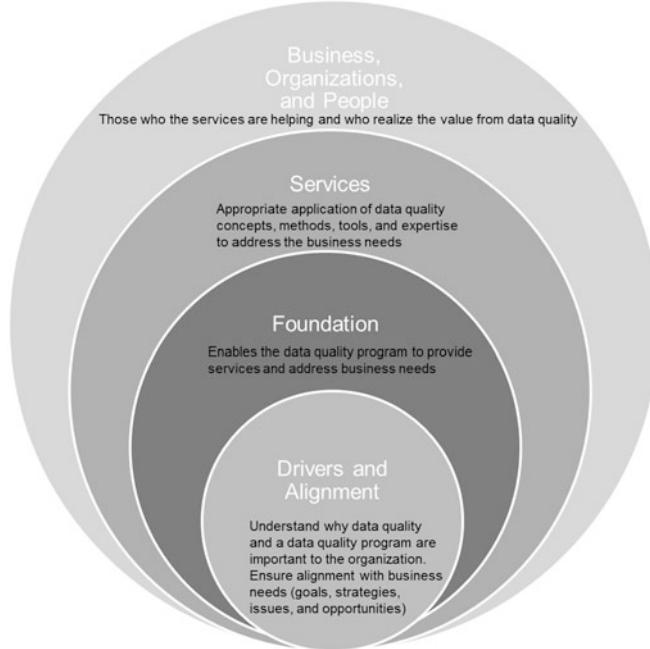


**Fig. 3** Approaches to data quality in projects and the Ten Steps™ process

over, the project team goes back to their home business units or they move onto the next project. Without the foundational elements that the data quality program provides there will be no expertise, tools, and resources to ensure data quality is appropriately included in projects. No one is responsible for training people in data quality concepts, methods, and techniques, so there is expertise for the next project. No one provides support for data quality tools, pays for licensing and maintenance, or ensures the tools are in place for other projects. When a new strategy is introduced to an organization, it is often necessary to create a separate program to kick it off. In order to move it from add-on work to the way you do business, it must be incorporated into the existing organizational infrastructure. While the discussion of where that function belongs is critical, it is outside of the scope of this chapter. There has to be a balance between the program and the projects because many data quality programs have ultimately disappeared because all the resources were put onto projects and attention to the foundation languished. If the foundation crumbles, there is no glue to hold everything together and eventually the projects and their results that were so valued also disappear.

## 5.1 Data Quality Program Framework

A Data Quality *Program* Framework shows the components necessary to include in a data quality *program*. It does not tell you how to implement those components



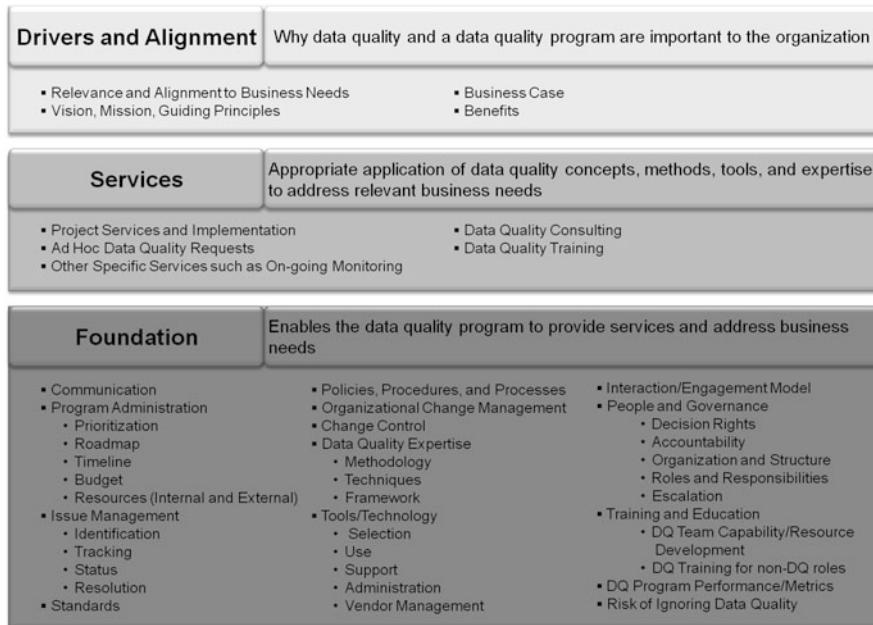
Copyright © 2010 Danette McGilvray, Granite Falls Consulting, Inc. All rights reserved.

**Fig. 4** Data Quality Program Framework relationships

but provides a visual checklist to see if you have addressed those components as you design your own data quality program. Later in this chapter we will show you how two different companies took the ideas from the framework and implemented their data quality programs. The relationships between the main sections of the DQ Program Framework are illustrated in Fig. 4. Starting from the center and moving outward:

- **Drivers and Alignment:** The reason for the program, what data quality means to your company, why DQ and the program you are developing are important, and the risk of ignoring data quality. All should be aligned with business needs. This is the starting point.
- **Foundation:** The foundation enables the DQ program to provide the services and address business needs.
- **Services:** Services are data and information quality in action. Services are the appropriate application of data quality concepts, methods, tools, and expertise to address relevant business goals, strategies, issues, and opportunities.
- **Business, Organization, and People:** Those the services are helping and who realize the value from data quality.

If any of the three inner circles are not addressed, then the business, organization, and people will not gain the value from data quality.



**Fig. 5** Data Quality Program Framework

Additional details of the elements within each section can be seen in Fig. 5. In its most simple form, there are three parts to the DQ Program Framework (see Fig. 5):

- Drivers and Alignment:** As mentioned, this section covers why data quality and a data quality program are important to the organization. This is the starting point for creating the data quality program and should guide decisions about which services to provide.
- Services:** The data quality services to be provided will vary depending on the needs of your organization. Examples are listed in Fig. 5.
- Foundation:** Elements necessary to sustain the ability to provide the services and address business needs.

If it looks like there is a lot to the foundation—there is. But this is no different than what is needed to manage any team, business unit, or program within any business. There are certain fundamental pieces that must be in place in order for the group to provide the products or services for which it is responsible. Each organization must look at the components in the DQ Program Framework and determine how they will implement the components and in which order. Note that the Ten Steps methodology introduced in the previous section addresses requirements relating to data quality expertise (methodology, techniques, and framework) in the program framework.

Having a strong ongoing program gives your organization the ability to carry out more successful projects because the methods, expertise, tools, and support are in

Domain	Description	Activities
<b>Information Quality Strategy and Governance</b>	This domain includes the efforts to provide the structures and processes for making decisions about an organization's data as well as ensuring that the appropriate people are engaged to manage information throughout its life cycle.	Activities include working with key stakeholders to define and implement information quality principles, policies, and strategies; organizing data governance by naming key roles and responsibilities, establishing decision rights; and building essential relationships with senior leaders in order to improve information quality.
<b>Information Quality Environment and Culture</b>	This domain provides the background that enables an organization's employees to continuously identify, design, develop, produce, deliver and support information quality to meet customer needs.	Activities include designing information quality education and training programs; identifying career paths; establishing incentives and controls; promoting information quality as part of business operations; and fostering collaborations across the organization for the purpose of engaging people at all levels in information quality strategies, principles, and practices.
<b>Information Quality Value and Business Impact</b>	This domain consists of the techniques used to determine the effects of data quality on the business as well as the methods for prioritizing information quality projects.	Activities include evaluating information quality and business issues; prioritizing information quality initiatives; obtaining decisions on information quality project proposals; and reporting results to demonstrate the value of information quality improvement to the organization.
<b>Information Architecture Quality</b>	This domain includes the tasks that assure the quality of the data blueprint for an organization.	Activities include participating in the establishment of data definitions, standards, and business rules; testing the quality of the information architecture to identify concerns; leading improvement efforts to increase the stability, flexibility, and reuse of the information architecture; and coordinating the management of metadata and reference data.
<b>Information Quality Measurement and Improvement</b>	This domain covers the steps involved in conducting data quality improvement projects.	Activities include gathering and analyzing business requirements for data; assessing the quality of data; determining the root causes of data quality issues; developing and implementing information quality improvement plans; preventing and correcting data errors; and implementing information quality controls.
<b>Sustaining Information Quality</b>	This domain focuses on implementing processes and management systems that ensure ongoing information quality.	Examples include integrating data quality activities into other projects and processes (e.g., data conversion and migration projects, business intelligence projects, customer data integration projects, enterprise resource planning initiatives, or system development life cycle processes); and continuously monitoring and reporting data quality levels.

**Fig. 6** Information Quality Certified Professional (IQCP) Domains. International Association for Information and Data Quality (IAIDQ). Source: <http://iaidq.org/iqcp/exam.shtml> [1]. Used by permission

place. Can you imagine a company telling each team to just figure out how they want to manage finances, develop their own chart of accounts, report financials whenever they feel like it, and hire anyone who knows how to use an online tax program to be their controller? Yet, that is often what happens with information when a company expects those with specific business or technology expertise to be able to fill in the skills of an information quality professional. “Just get a good data entry person and our data quality will be fine” is a phrase often heard. But data quality requires much more than data entry.

## 5.2 *Information and Data Quality Skills*

Information/data quality is a distinct profession [5]. IAIDQ, the International Association for Information and Data Quality, provides a certification, Information Quality Certified Professional (IQCP<sup>SM</sup>) which covers a wide range of topics, within six major areas of knowledge (domains) considered essential to an information quality professional. For a list and description of the domains, see Fig. 6. The



**Fig. 7** DMBOK Functional Framework. Source: *DAMA-DMBOK*, p. 7 [3]. Used by permission

associated 29 tasks and several hundred distinct knowledge and skills show the expertise in and practical knowledge of information quality principles, concepts, and methods needed to implement information and data quality. As a data quality program manager, you can use these to determine skills and training needed for your data quality team, write job descriptions, and plan your program.

In addition to working with technology and the business, your data quality team needs to collaborate with other areas of data management. DAMA International has developed the Data Management Book of Knowledge (DMBOK) and Functional Framework [3] which shows their view of how these aspects work together (see Fig. 7). Depending on your organization, the data quality program could be an umbrella structure for some of these other data management areas or it could be under the umbrella of another. In any case it is important to understand that information quality has its own body of knowledge and skill sets. One of the responsibilities of the data quality program is to determine the interaction and engagement with other parts of the organization and to ensure that people have the proper training, knowledge, experience, and skills to carry out data quality work.

## 6 A Tale of Two Companies

Drawing on the background and basic framework of components needed for a data quality program, let's see how two companies have implemented a formalized data quality management program and are executing data quality projects. Remember these are real companies, with real data and information quality challenges that impact their businesses. Both have been successful with their approach to data and information quality.

### 6.1 *Company A*

#### 6.1.1 Company Background

Company A is a US financial services company that was established almost 40 years ago. It manages billions of dollars worth of loans and serves 23 million customers. It offers a variety of products and services designed to help customers manage their financial needs. A Fortune 500 company, it has over 6,000 employees in offices throughout the USA.

#### 6.1.2 Data Quality Program Background and Timeline

Company A appointed a Director of Enterprise Data Management and the Data Governance Office. The director and her team were responsible for the successful design and implementation of the enterprise data governance and data quality programs at the company. In her words, here is the background of the work they did leading up to the formation of the enterprise data quality program. Dates are included because it is helpful to see the time it takes to execute projects and build programs.

“Our company began formalizing a data governance program in the beginning of 2006. As a first step in the spring of 2006 we kicked off a project to identify our enterprise data. With the data we identified, we created a list of enterprise fields that we utilized as the scope of our future Data Governance Program. Our next phase was a pilot project that focused on addressing the issues of seven key fields needed to improve the marketing efforts within our company. The identified roles and responsibilities, lessons learned, and the issue resolution framework used during this project were utilized as input to the development and design of our Data Governance (DG) Program. We worked with Gwen Thomas of the Data Governance Institute from November 2006 until we went live with the DG program in April 2007.”

“In the fall of 2009 we started formalizing an enterprise Data Quality (DQ) Program that fits under the Data Governance umbrella. The program was designed and the first pilot project implemented with help from Danette McGilvray of Granite

Falls Consulting, Inc. We utilized her approach to data quality called Ten Steps to Quality Data and Trusted Information™ (Ten Steps™). We conducted a pilot DQ project that resulted in our company successfully implementing 22 high priority business rules and a DQ Dashboard that reports on the following: 1) State of Data Quality using red, amber, and green statuses with the ability to drilldown to historical trends and other detail, 2) Business Value that includes the total projected and actual amounts for revenue generated and costs avoided, as well as intangible benefits of data quality, 3) DQ program performance by the number and status of DQ Issues and DQ Engagements.”

“The formal, enterprise Data Quality Program went live in June 2010. The Data Quality and Governance Programs continue to work closely together. Additional business rules are monitored according to business priority. The DG Program provides guidance, prioritization, and decisions for DQ activities and oversees resolution of DQ issues. The DQ program develops DQ management as a core competency and improves DQ throughout the company. Results from the Data Quality monitoring have been used by the business to improve its consumer marketing strategies, decrease cost of funding facilities, and streamline origination and servicing practices while ensuring that risk and compliance issues are managed and controlled.”

### 6.1.3 DQ Program Plan

The DQ program plan was called the Data Quality Cookbook, which mirrored the Data Governance Cookbook developed during the data governance program implementation. It is the plan and roadmap for data quality at Company A and was used to institute a formalized data quality program within the company. It was developed by a team of six people over the span of 3 months. The cookbook was designed to be a reference for the staff of the Data Governance Office (DGO) and Data Quality Services (DQS) during the implementation and operation of the data quality program. The DQ Cookbook and the processes it represents were not intended to supersede established IT disciplines and skill sets such as data governance, metadata management, or data architecture. It was also not intended to suggest that data quality tasks typically addressed by these disciplines or others within the company fall under the purview of DQS. It was, however, intended to show that DQS can help facilitate alignment of DQ activities between such disciplines and participants.

The end result of the planning was the Data Quality Cookbook which was made up of 11 modules:

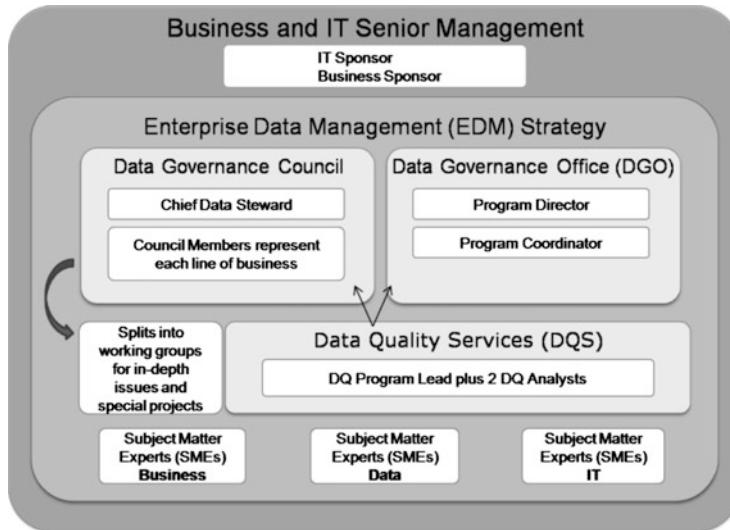
- **Vision, Mission, Guiding Principles, and Policies:** Why the data quality program exists, what we want to be, and principles and high-level policies that drive our decisions and actions and determine what will be put into practice.
- **Benefits:** Benefits from data quality and from having a formalized DQ program.

- **Organization, Roles, and Responsibilities:** The Data Quality Services (DQS) organization, reporting, and team structures; the relationship between DQS, DGO, and other groups within Company A; key roles and responsibilities; and required skills and knowledge.
- **Services and Engagement Model:** The components of the DQ program; DQS and their engagement models; DQS program management; and DQS-owned tools.
- **Metrics:** The data quality metrics to be implemented by DQS, techniques for prioritizing data to monitor; background on metrics, and examples from within Company A of “why” to do metrics.
- **Framework and Methodology:** An overview of the data quality framework and methodology chosen by Company A as the standard process for creating and improving data quality. This is Ten Steps to Quality Data and Trusted Information™ (the Ten Steps™ methodology) developed by Danette McGilvray, Granite Falls Consulting, Inc.
- **Technology:** Tools used by DQS, technology environment, integration between the modeling and data quality tools, and tracking database for DQ issues.
- **Training:** Current and future DQ training options for DQS customers and for DQS and DGO internal resources (train the trainer and capability development) for various delivery methods (classroom, one-on-one, documents, community).
- **Communications:** Communications objectives and approach; key messages, communication plan, audience analysis, and initial DQ communications.
- **Implementation Roadmap:** Next steps for pilot and high-priority activities; results of prioritizing the activities, initiative, and engagements DQS should be involved.
- **Business Value Anecdotes:** A repository of various data quality anecdote documents by the DQS and DGO teams; template; and additional background on building the business case for data quality.

Implementation of the handbook started the following quarter with training on the Ten Steps methodology. The first project was to design and implement an ongoing process for monitoring, reporting, and addressing high-priority data within the company.

#### 6.1.4 Organizational Fit

Company A’s data quality program is part of the Enterprise Data Management (EDM) Strategy. Implementing a formalized DQ program had been part of the company’s long-term plan. Before embarking on the data quality program, Company A had spent 15 years building out and maturing the first three components of the EDM Strategy (metadata management, data architecture and design, and data management services definition) using a centralized, enterprise data management organizational structure and approach. They built a centralized data management team that implemented an enterprise data modeling tool/enterprise data dictionary.



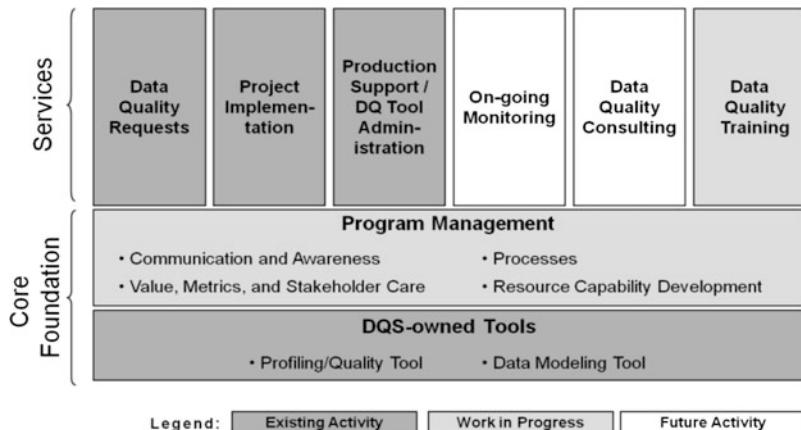
**Fig. 8** Company A: DG/DQ program organizational structure

This team provided all the technical and business metadata for all the IT-supported systems. They devoted time to implementing and maturing all the data management services, including database administration (DBA), operational, backup/recovery, and security.

During the DQ program planning phase the initial DQ project team determined exactly how DQ would fit and how it would interact with the data governance program and created the Data Quality Services team. Figure 8 shows how the data quality program (Data Quality Services—DQS) is under the umbrella of the data governance program and both are part of the Enterprise Data Management Strategy. The Data Governance Office and Data Governance Council report to upper management through an operating committee made up of senior management on the IT and business sides of Company A. This committee is responsible for prioritizing and approving all business-driven initiatives.

Issues can be identified by the Data Governance Council, project teams, anyone in the business, IT, or from management. The Data Governance Office (DGO) works with DQS and other subject matter experts (SMEs) to research the issues. Results of the research and recommendations are provided to the DG Council, which is made up of representatives from all the lines of business. The council and DGO make decisions and take action, including DQS where applicable. Larger issues may require the formation of working groups while the issue is investigated and resolved.

The Data Quality Services team administers the data quality program and provides services for data quality requests, project implementation, production support/data quality tool administration, ongoing monitoring, data quality consulting, and training. This group currently consists of three individuals and they report to



**Fig. 9** Company A: DQ program components

the Data Governance Director. Working directly with the stewards, they identified Business Rule Approvers who are accountable for each of the existing 96 business rules that are being monitored on a weekly basis.

Company A won two prestigious awards for data governance due in part to its data quality program. One of the important judging factors in both competitions was the ability to demonstrate business value for the organization.

### 6.1.5 DQ Program Components

Company A defined the services that the DQ program would provide and the core foundation activities that have to exist in order to provide the services. Some they had been providing already on an ad hoc basis and others were services they wanted to provide. When the program first went live you can see that some activities already existed, some were in progress, and others planned as future activities would be addressed during the first pilot project (see Fig. 9). As of writing this chapter, all of the program components have been implemented.

### 6.1.6 Initial Project

Once the DQ program was planned, the first project implemented ongoing monitoring of key data as prioritized by the Data Governance Council. This included instituting a data quality dashboard (see Fig. 10). Three categories of metrics are reported: (1) State of Data Quality, (2) Business Value from Data Quality, and (3) Data Quality Program Performance. For each category, a dashboard-level status is summarized and detailed reports are provided. Drilldown information is included as appropriate for the specific metric, such as trends, detailed statistics,

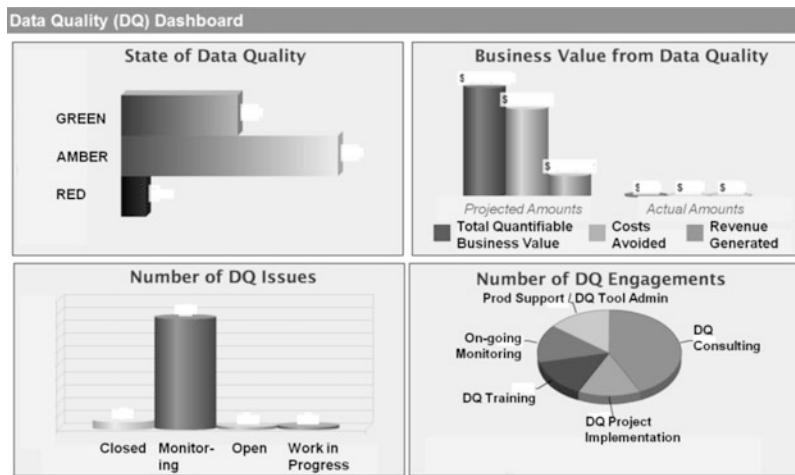


Fig. 10 Company A: data quality dashboard (actual numbers concealed for confidentiality)

and definitions. The State of Data Quality metrics are based on business rules that were approved by the Data Governance Council and are written into their data quality monitoring tool. The monitoring runs automatically each week and the dashboard is updated. The calculations for the Business Value from Data Quality were developed using business impact techniques from the Ten Steps methodology. These calculations are applied against the data quality monitoring results and are updated monthly. The Data Quality Program Performance is based on the issue log and other work that the Data Quality Services team provides and is updated monthly. DQ metrics continue to be added on a regular basis.

The Business Rule Approvers mentioned earlier are accountable for the results of the monitoring and ensuring appropriate actions are taken if a business rule has a red or amber status. They sign off on all information needed for the State of Data Quality and Business Value Metrics. They review monitoring results and determine if there is really a problem and if additional action is needed. They assign resources, as needed, to identify and fix root causes of the data quality issues, correct data errors, update the monitoring process, and verify results in the next monitoring run. The Business Rule Approvers may delegate work to others, but must answer that the work is accomplished and are ultimately responsible.

DQ Program Performance Metrics show activity in the following areas:

- DQ issues
  - Data quality requests (closed, open, work in progress)
  - Ongoing monitoring (reflects the “State of Data Quality” business rules/metrics being monitored on a regular basis)

- DQ engagements
  - DQ consulting
  - DQ project implementations
  - DQ training
  - Production support/DQ tool administration
  - Ongoing monitoring

### 6.1.7 Results

The following project is just one of many where data quality has helped Company A.

**High-Quality Reports** Company A develops monthly reports for its clients who use them to ensure continued compliance with federal regulations. These reports provide a fee-based revenue stream for Company A. Data from the report system also feeds into Company A's collections systems for new placements, to be used by collection agents.

The Data Quality Services (DQS) team was engaged to provide a robust data quality architecture to help alleviate some data quality issues. The DQS team implemented an automated reusable file automation architecture using their profiling/monitoring tool that analyzes the data quality of the incoming client files every 15 min ( $24 \times 7$ ). The reusable file automation architecture pushes the detailed data quality metric information to both the report management and production support staff for each incoming client file. The pushed information provides enough insight to the staff to help them understand if they have all the information needed to run the reports for their clients. If needed, the information also allows the production support staff to engage a client about a data quality problem in a more expedient manner. Managing this combination of the right processes, people, tools, and data provides a stable foundation that built confidence in the report service provided to their clients, increases the management visibility and quality of their reporting, and improves the data quality going into and out of the report-generating system. All of this benefitted Company A's clients.

**Other Projects** In other areas of the company, DQS has instituted reconciliation checks between source system feeds and the enterprise data warehouse and quality checks on deals of loans they purchase before loading them onto their loan acquisition system. Equally important, they are working closely with data governance to ensure the appropriate resources are in place to address issues and put measures into place to prevent issues from arising again.

Increasing revenue, managing cost and complexity, and supporting compliance initiatives are all high-priority business needs at Company A. Data quality helps meet those business needs in the following ways:

- Increase Revenue. Supports retention of borrowers and supports marketing initiatives through better customer segmentation, targeting its best customers, and marketing in a timely fashion.

- Manage Cost and Complexity. Supports more efficient and effective spending around marketing campaigns; helps Company A minimize losses from loan defaults; avoids costs of ad hoc efforts to identify, assess, and correct data quality issues; supports better revenue forecasting and debt management forecasting; better calculations used in Premium Models; better calculation of required Loan Loss Reserves; and reduces reconciliation efforts caused by redundant data and multiple, disparate sources of data.
- Support Compliance Initiatives. Helps maintain compliance with privacy requirements and marketing constraints and improve global understanding of data across lines of business.

## 6.2 *Company B*

### 6.2.1 Company Background

Company B is an automotive retailer in the United States operating many dealerships spread across nearly a third of the USA, including many major metropolitan markets. These dealerships sell new and used cars, light trucks, and replacement parts; perform vehicle maintenance, warranty, and collision repair services; and arrange extended warranty contracts, financing, and insurance. The vehicles sold represent a variety of different import, luxury, and domestic automotive brands.

### 6.2.2 Data Quality Program Background and Timeline

Historically, Company B's strategy was to buy and build a portfolio of car dealerships, which operated fairly independently. Corporate reporting needs were met with spreadsheet reporting and third-party-hosted systems. In 2008 a data management team was formed to implement data warehouse and business intelligence (BI) applications with an initial focus on getting consolidated data in a timely manner. The BI team recognized that data was an important asset to the company and had a vision of developing a data-driven culture for decision-making, with one source of truth for enterprise data. They wanted to provide self-service ad hoc query capabilities, support predictive analytics, and deploy via the web and mobile devices. As data from the dealerships was integrated into the warehouse, the data management team needed to understand the data and saw that data quality problems were causing difficulty in using the data.

At the same time, Internal Audit was also recognizing the need for data quality. In 2009, Internal Audit developed a business plan to adopt a risk-based auditing method and created the Audit Analytics team. One of the auditors teamed up with the data warehouse architect from the BI team to focus on data. In 2010, BI and Audit Analytics interviewed every business area within the company to gather information about how they currently use data to manage their business

area, including asking, “What would you do with additional data if you had it?” Standardization/quality issues were a popular theme among business groups when discussing reporting challenges. In 2011, the enterprise data warehouse was redesigned and a renewed understanding of the need for data quality emerged. Some of the issues included challenges with source system data due to unreliable data acquisition and no referential integrity. Inconsistent processes across the dealerships resulted in inconsistent data.

The auditor and the data warehouse architect, who were now working closely together, decided to move forward in their data quality efforts by soliciting executive support. They presented their concerns to the Chief Financial Officer (CFO). Armed with real examples of issues from the company’s data, he was able to understand the problem. They then created a one-page document summarizing the issues which the CFO used to create awareness among senior management. Executives approved a data quality project with senior management sponsorship and a project team. Unfortunately, this first project failed due to resource constraints, lack of experienced data quality professional resources, and poor alignment with other data initiatives. While they had executive approval they found it was not enough—they needed executive *involvement*.

This realization proved to be a turning point: as the auditor and the data warehouse architect assessed what was missing in the failed project, they realized data quality was not just a project. They began thinking of data quality as a service to be included in all projects. When starting research about the data quality discipline they discovered a data quality conference (DGIQ—Data Governance Information Quality) was only a few weeks away. They attended the conference and absorbed as much as possible while discussing program ideas with presenters, other attendees, and consulting professionals. In these discussions they identified what was needed to refocus their efforts and build a long-term data quality program.

Once back home at Company B, the pair identified projects within the company that relied on high-quality data and showed management the total investment in current and planned data-intensive projects that was at risk. There were several key projects underway that required high-quality data to succeed, such as using data to understand the company’s customers. This understanding would help drive strategic marketing initiatives and aid in reaching the company’s overall business objectives. They also showed management the number of resources dedicated to application development compared with the lack of resources dedicated to data quality within those projects.

They created a draft business plan and obtained support from the CFO, CIO, and Chief Executive Auditor to approach data quality as a *program*. Their approval included dedicating the majority of the original internal auditor’s time to managing the data quality program. The company was now ready to formalize a data quality program and, in the words of Company B, “establish an Enterprise Data Quality culture where quality information is achieved, maintained, valued, and used to enable business strategies through a partnership between the business and technology.”

They engaged an external consultant, Danette McGilvray, to provide an assessment of the draft business plan. The first (1-month) engagement resulted in a review and assessment of the existing high-level business plan, relevant company documents, and a one-week on-site visit. Because of their preparation, the on-site visit was less of an assessment and more of actually “doing the work” to formulate the data quality program and start putting it into place. During the on-site visit they finalized the DQ structure, roles, and responsibilities; developed the program framework; and extended the existing high-level 6-month roadmap to include Phase 1 tasks and a timeline (going out 12 months). They also created the initial communication plan, developed high-level procedures for addressing ad hoc data quality requests, and obtained executive support for the program plans.

### 6.2.3 DQ Program Plan

Company B’s data quality program plan outlined the components necessary to achieve a high level of data quality across the company. The plan included an executive summary; the overall objective of the program; the program framework and services, structure, roles, and responsibilities; and interaction model. It discussed what was already in place as well as what was needed to be developed in the way of process, technology, and communication. It outlined the program phases, timeline/roadmap, critical success factors, risks, and budget needs.

Those involved understood that the data quality program would evolve and be adjusted based on the ever-changing business and technology environments. Management agreed that dedicating the right people, process, and technology to managing data as a valued corporate asset would enable good business decisions and innovation that would allow Company B to stay ahead of competitors.

Company B’s high-level strategy for its data quality (DQ) program was to:

- Formalize a data quality team to provide DQ Services to projects prioritized by data management and the CIO
- Ensure appropriate people would be trained to provide DQ Services or to incorporate DQ activities into their daily responsibilities
- Communicate relevant data quality messages to appropriate audiences across the enterprise
- Ensure data quality would be a part of the specific business process guidelines for each end user application used by the dealerships, referred to as “playbooks.” The applications addressed by the playbooks were the main source of data used by business intelligence and the data warehouse
- Improve the State of Data Quality within the data warehouse
- Prevent data quality issues whenever possible (NOT just find and fix data errors)

The plan also laid out areas of low risk and high risk in order to institute the program. The program had a higher probability of successful implementation because of reasonable and clearly defined program objectives and scope and committed sponsors. There was funding available to hire two data quality analysts, purchase a data quality tool, and engage the external consultant as needed through

<b>Drivers</b>	<b>Why data quality and a formalized data quality program are important to our company</b> Relevance, Benefits, Business Case, Vision and Mission, Alignment		
<b>Services</b>	<b>Appropriate application of data quality concepts, methods, tools, etc. to solve problems</b> Project Services, Ad Hoc Requests, Existing DQ Issues		
<b>Foundation</b>	<b>Enables the program to provide services and address business needs</b> <ul style="list-style-type: none"> <li>• Decision Rights</li> <li>• Accountabilities</li> <li>• Structure</li> <li>• Roles and Responsibilities</li> <li>• Escalation</li> <li>• Prioritization</li> <li>• Playbook</li> <li>• Communication</li> <li>• Interaction Model</li> <li>• Training</li> <li>• Policies</li> <li>• Program Administration</li> <li>• People</li> <li>• Tools</li> <li>• Budget</li> <li>• Change Control</li> <li>• Organizational Change Management</li> <li>• Support</li> <li>• Methods</li> <li>• Processes</li> <li>• Standards</li> <li>• Program</li> <li>• Performance/Metrics</li> <li>• Risk</li> <li>• External Consulting</li> <li>• Issue Management</li> <li>• Project Management</li> <li>• Vendor Management</li> </ul>		

**Fig. 11** Company B: Data Quality Program Framework

the first phase. The highest risk area lay in the number of dependencies this program had on other projects in the company and the fact that there was a rapidly increasing amount of data entering the business on a daily basis. Other high-risk areas included continuously changing business processes, expectations, source systems, and rules. These would need to be clearly defined and updated by the business so they could be correctly applied against the data. All risks would have to be constantly identified and evaluated over time.

#### 6.2.4 Data Quality Program Framework

The DQ Program Framework was developed during the first 1-month consulting engagement (see Fig. 11). This framework provided an easy reference for the data quality team to understand the components necessary to have high data quality and a successful formalized data quality program at Company B:

- **Drivers** indicated why data quality and a formalized data quality program were important to Company B. It was critical that *any* data quality work be associated with a business need, goal, issue, opportunity, or strategy. (The drivers were articulated in the business plan.)
- **Services** were the appropriate application of data quality concepts, methods, tools, etc. to solve problems. The initial services to be provided by the data quality program were
  - Project Services: Incorporating relevant data quality activities into projects.

- Ad Hoc Requests: Addressing data quality-related requests that can come at any time, from any area of the company.
- Known Issues: Addressing data quality issues that were already known at the time. In practice, the ad hoc requests and known issues were merged into one list and addressed according to priority.
- Note: It was already clear that the known projects, requests, and issues outpaced the resources available. Good prioritization was vital.
- The **foundation** enabled the program to provide services and address business needs. Note that all the foundational elements were important, but all did not have to be implemented at once. The elements could be implemented over time at the appropriate levels of detail to meet the company's needs.

### 6.2.5 Organizational Fit

The program's executive sponsors were the Vice Chairman and CFO and the CIO. Key stakeholders were the Chief Audit Executive, the Director of Enterprise Business Intelligence, and the Director of Business Applications.

The DQ team was a cross-departmental team. The original internal auditor was named the new data quality program manager and continued to report up through the Chief Audit Executive, who was fully supportive of the program. The rest of the team (the original data warehouse architect and two newly hired data quality analysts) were part of Business Technology and reported to the Director of Enterprise Business Intelligence. (Note that it was not "Information Technology" but "Business Technology," a distinction that has been made purposefully at Company B because the technology group aligned closely with and enabled business strategies.) The DQ team also worked closely with three business analysts and an ETL developer, who could have been considered an extended DQ team. The key roles related to the data quality program at Company B were visualized in the data quality structure (not included here). The structure showed the virtual team just described, but was NOT an organizational reporting chart.

Management agreed on the stated roles and responsibilities related to the data quality program (see Fig. 12). You can see the interaction of these roles in Fig. 13.

### 6.2.6 Data Quality Phase One

Goals for Phase 1 of the program (0–12 months) included both Services and Foundation aspects of the program:

#### Services

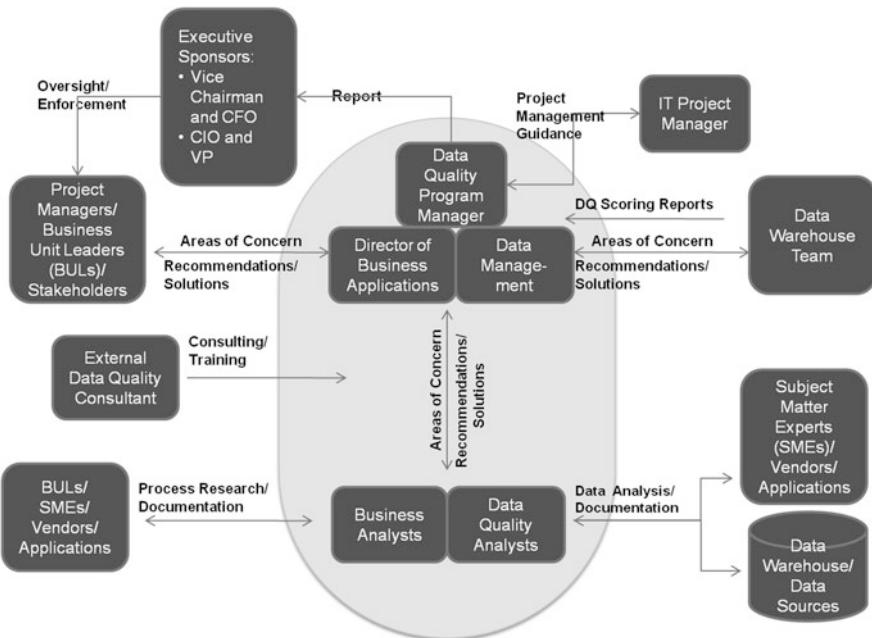
- Project Services: DQ team would incorporate and implement data quality procedures for the highest-priority data-intensive projects.
- Ad hoc Requests: Apply DQ procedures to prioritized ad hoc requests and known issues.

DQ Program Role	DQ Program Responsibilities
Executive Sponsors	<ul style="list-style-type: none"> <li>• Executive oversight</li> <li>• Ensure enforcement of decisions</li> <li>• Facilitate agreement when needed</li> <li>• Aware of impacts of poor quality data and benefits of improving quality and creating a quality culture</li> <li>• Defend and enforce change when necessary</li> </ul>
DQ Program Manager	<ul style="list-style-type: none"> <li>• Oversee all activities to ensure that the objectives of the DQ Program are met</li> </ul>
Project Manager SME	<ul style="list-style-type: none"> <li>• Provide guidance on project management practices to follow Project Management Office standards (organizational, accounting, billing, actuals to budget, etc.)</li> </ul>
Stakeholders	<ul style="list-style-type: none"> <li>• Facilitate DQ program</li> <li>• Clear roadblocks</li> <li>• Address priorities</li> <li>• Responsibility and authority to ensure processes are effectively executed.</li> <li>• Publicize the responsibility – as well as the penalties for noncompliance</li> <li>• Communicate known issues to the DQ program</li> <li>• Closely align with business analysts and data quality analysts to prioritize issues and ensure decisions about data quality meet the requirements of the line of business</li> </ul>
Data Management	<ul style="list-style-type: none"> <li>• Determine data needs/issues</li> <li>• Prioritize data quality work</li> <li>• Manage the data quality analysts</li> <li>• Member of core data quality team</li> <li>• Work closely with Business Applications to ensure coordination of data quality analysts and business analysts</li> </ul>
Data Quality Analysts	<ul style="list-style-type: none"> <li>• The experts in Data Quality</li> <li>• Technical knowledge of data profiling and cleansing, as well as a good understanding of data structures, relationships, models, and requirements</li> <li>• Manage metadata related to data quality</li> <li>• Strong communicators and problem solvers who work well in a team environment</li> <li>• Work closely with the business analysts to ensure decisions are made that satisfy the data and technical requirements of the applications as well as the business and their information needs</li> <li>• Lead the development of source to target mapping and gather data requirements</li> <li>• Identify root causes and implement measures to prevent data quality problems</li> </ul>
Business Applications	<ul style="list-style-type: none"> <li>• Manage business analysts</li> <li>• Work closely with Data Management to ensure coordination of data quality analysts and business analysts</li> </ul>
Business Analysts	<ul style="list-style-type: none"> <li>• Work with Business Unit Leaders to implement technological controls and operational processes to ensure data quality</li> <li>• Owners of the applications, resolve issues</li> <li>• Experts in applications, processes, and the information that supports the business needs</li> <li>• Strong knowledge of the business relationship between data and how it relates to business processes</li> <li>• Craft the appropriate data definitions and business rules</li> <li>• Ensure business rules and data rules are fit for the needs of each line of business</li> <li>• Work closely with the data quality analysts to ensure decisions are made that satisfy both their understanding of the business and the technical requirements of the applications</li> <li>• Understand what kind of data stakeholders need and how they use it</li> <li>• Responsible for business process mapping and participate in the data requirements analysis process</li> <li>• Identify root causes and implement measures to prevent data quality problems</li> </ul>
Data Quality Consultant	<ul style="list-style-type: none"> <li>• Provide overall guidance and consultation on how to develop and implement an enterprise-wide DQ program</li> <li>• Provide data quality training</li> </ul>
Business Unit Leaders (BULS) or Appointee	<ul style="list-style-type: none"> <li>• Decision makers on standards and processes to ensure DQ objectives are met without impacting operations.</li> <li>• Will only participate when their area is being implemented</li> </ul>
Subject Matter Experts (SMEs)	<ul style="list-style-type: none"> <li>• Assist with area of expertise</li> <li>• Provide expertise in reporting needs, data quality issues, understanding business, what information is important, how want to see data</li> <li>• Assist with root cause analysis and implementation of measures to prevent data quality problems</li> </ul>

**Fig. 12** Company B: DQ program roles and responsibilities

## Foundation

- Formalize DQ Program Framework.
- Develop program structure, roles, responsibilities, and interaction model.



**Fig. 13** Company B: DQ interaction model

- Train DQ team on current environment, applications, and data quality concepts, methods, and tools.
- Select and implement appropriate DQ tool(s).
- Complete DQ procedures for project services, ad hoc requests, and known issues.
- Incorporate data quality activities into the SDLC used by Company B.
- Communicate DQ program to executive sponsors and Business Unit Leaders.
- Develop DQ Program SharePoint site and processes for tracking data quality activities.

Phase 2 (13–24 months) plans for services were to evaluate progress and set goals to continue to move DQ maturity from reactive to proactive. Foundation plans included refining foundational elements from Phase 1 to be repeatable and sustainable, instituting progress reports using the processes implemented in Phase 1, and defining DQ metrics.

As of the writing of this chapter, the DQ team was partway through Phase 1. They completed training of the data quality team, business analysts, and some of the data warehouse staff on (1) the current environment of Company B, systems, and data; (2) specialized training in data quality techniques and processes—the Ten Steps methodology; and (3) on the DQ tool once it was purchased (see more detail below). Phase 1 included incorporating DQ activities into current projects and addressing prioritized DQ requests. Everything learned in Phase 1 was meant to

evolve into repeatable processes, which would lower the cost of future DQ projects and advance the maturity of the program toward the goal of data quality across the enterprise.

### 6.2.7 Initial Projects and Results

Remembering that I use the term *project* very broadly, let's look at the initial projects that Company B completed or was in the process of completing during Phase 1 of the data quality program plan in both the foundation and services areas.

**Data Quality Tool Selection (Foundation):** The first foundational project for the team was to select a data quality tool. Their initial focus was on profiling and discovery functionality, though they also looked at cleansing and standardization. A rigorous selection process was completed where they first identified their selection criteria—the functions and features most important to them. For example, because they had not closely analyzed their data, they needed the strong ability to profile their data, discover relationships, and find out “what they didn’t know they didn’t know” about their data. They also needed to take results from any profiling tool and to visualize these results through the business intelligence tool used by the company. Phone interviews, demos, RFIs (requests for information), and finally proofs of concept for the finalist vendors completed the selection process.

**Initial Quick Win:** DQ work played a critical role in the release of one application playbook and related dealer training. The Director of Enterprise Business Intelligence sent an email to the executive sponsors and another project stakeholder (the Senior Vice President and Corporate Controller), acknowledging the importance of data quality work to the success of the rollout. In his own words: “The data quality aspect of this project was in the playbook questions. If the application had questions that didn’t make sense, it would have reflected poorly on the app just as much as finding a bad number in a set of financials throws the rest of the financials into question. One of our new data quality analysts poured over the playbooks, validated each question back to the playbook, corrected simple errors, made sure the scoring was accurate, clarified discrepancies, identified valid answers, and then made additional modifications when the Business Unit Leaders or auditors needed to make changes. As a result, when the app was used, all playbooks were in much better shape because of her data quality focus and efforts than they would have been had we just been thinking about the technology. I might add there were very few playbook questions that she did not have any changes or clarifications that needed to be made. In the end, the roll-out went smoothly and the feedback from the auditors has been extremely positive.”

**Data Flow Integrity Project:** An in-progress project carried out by another data quality analyst dealt with understanding the information life cycle and foundational data that touched several projects and operational processes. This project included analyzing and instituting solid processes to prevent data problems along with monitoring and validating data movement to quickly identify and react to problems if they occur.

**Project Services:** This service centered on incorporating appropriate data quality activities into several current and upcoming projects. The executive sponsors prioritized and narrowed to 5 those projects where DQ Services would initially be provided. All these projects were aligned with strategic business initiatives. The data quality team then evaluated each project from a Ten Steps process perspective and determined which data quality activities should be included. For example, one of the DQ activities involved using the recently purchased discovery and profiling tool to understand the data and its condition. This understanding helped inform decisions on where to spend time on the data itself throughout the project. The DQ activities in every step of the project life cycle will prevent many problems from occurring once in production and keep business running smoothly.

## 7 Comparing the Companies' DQ Programs and Projects

Let me emphasize that both Company A and Company B have successful data quality programs. Both contain the fundamentals necessary to implement data quality projects and sustain their data quality programs even though their approaches differed.

Both companies had been doing DQ work on an ad hoc basis and realized the need for a formal data quality program in order to move their DQ efforts forward. Both brought in outside expertise to set up a DQ program by working with in-house employees with specific knowledge of their organization and data. Company A's DQ Cookbook outlined in detail the foundational elements of their program. Company B acknowledged the foundational elements of the program in their DQ Program Framework and will implement some of those elements more gradually. Both also recognize communication and training as important foundational components.

Company A's data governance program had been in place for about 3 years when work on the formal data quality program was started. The DQ team was organized under the umbrella of the Data Governance Organization. Company B understood that governance was important, but the word "governance" was not something they felt would be accepted in their environment at that time. However, the things that governance provided such as decision rights, accountability, structure, roles, responsibilities, escalation, and prioritization were included in the DQ program foundation, as these must be addressed. So in practice governance was under the umbrella of data quality. In other companies I have seen data quality and data governance as separate "sister" teams at the same level within the organization. The learning from this is that any of these organizational structures can be successful.

Both companies have management support, with Company B's support coming from a higher level in the organization than Company A. Company A and Company B have completely different overall business structures so it is natural that support would be coming from varying levels of management. However, both have the necessary support that allows them to move forward with their programs. As with any organization, both companies have to be prepared to sustain

support. They will be required to reeducate when there are changes to executive sponsors, stakeholders, and other contributing to DQ due to natural and ongoing shifts in the organization, roles, and responsibilities.

The DQ program in both companies offer the DQ Services most needed by their organization. Both companies started their data quality work as prioritized by the business side of the organization. Company A's first project was to implement ongoing DQ monitoring and metrics for critical data as prioritized by their Data Governance Council. The first project also included a DQ dashboard. Company B's initial focus was to incorporate DQ activities into projects as prioritized by executive and middle management. Company B plans on establishing DQ progress reports as part of Phase 2.

Both companies made good use of the existing company infrastructure and stayed consistent with what other teams in the company were utilizing. Company A developed their intranet website and their own issue tracking database. In Company B, a third-party collaboration tool was a company standard for content management for sharing program and project documentation. Another third-party tool was being implemented for project management at the time the data quality program began. Company B made use of both in their data quality program.

Company A already had a third-party DQ tool that had been in use for a few years to research and understand data for DQ-related requests. Company A's first project extended the use of the DQ tool by using it for the ongoing monitoring and reporting of key data fields. Company B had looked at their data using tools such as SQL and Toad but had no data quality or profiling tool per se. After establishing the DQ program, Company B first selected a data quality tool, which was then available to use as needed in their high-priority projects.

## 8 A Few Final Words

You have seen here the foundation behind executing data quality projects and the development of data quality programs. You have seen how two different companies have taken the foundation and applied the framework and methods to build a data quality program and execute data quality projects to meet their organization's specific business needs, environments, and culture.

Sustaining data quality within any organization requires both the ability to execute data quality projects and to have a foundational program that provides the support, tools, training, and resources (people, money, expertise) for ongoing data quality needs. Organizations of all sizes and types can implement successful data quality programs and projects. To do so well, they must understand and fund the foundational elements necessary for a data quality program and take into account the organization's environment and culture. They must have the ability and resources to execute data quality projects and provide other DQ-related services. Without the program, data quality work is at risk of not continuing. Without the data quality projects, data quality will not be actually implemented within the company.

## 8.1 Your Starting Point

What will be the starting point for *you* in *your* organization? Wherever you begin, I am very practical in the sense that you can only start where you are and where there is a need. Where are the complaints about data quality coming from? What are your organization's business needs, issues, goals, strategies, and opportunities and the associated data, processes, people and organizations, and technology? Where do you have support for data quality? If you are part of a data warehouse team and don't have personal access to the CEO, then you can't start with the CEO. However, you *can* start with the business manager in charge of business intelligence whose reports depend on the warehouse, the data warehouse IT manager or the enterprise architect. Gain support from your manager and colleagues, show success, and continue to increase support up the management chain. If you are fortunate to have access to any of your executive leadership team, then obviously start there, knowing that even with executive support you will still have to put effort into raising awareness and gaining support with those in middle management, individual contributors, and project managers and their teams. An executive mandate is *extremely* helpful, but on its own does not assure action from everyone else. I am confident that you will be able to do *something* related to data quality that will benefit your organization from wherever you are and with whatever data you are looking at today.

## 8.2 Cautions

As you put your plans together and implement your data quality program and projects, let me point out a few important cautions:

- While there will be a pull to give the new services and projects precedence, ensure you also build the foundation for the program. Let me reiterate what I said earlier, many data quality programs have ultimately disappeared because all the resources were put onto projects and attention to the foundation languished. If the foundation crumbles, there is no glue to hold everything together and eventually the services and their results that were so valued also disappear.
- Be assured that your projects, requests, and issues will outpace the resources available. This will not change so become proficient in prioritizing. Put the resources you have been given to full use. As time goes on and the data quality program shows value, expect to have even more requests for help. That will be the time to ask for more resources.
- Track progress and show value from what the data quality program and projects provide. There will always be someone who will need to see it. Even if you have support from management and others today, there will be an organizational change and you may have to start all over again with new management who is unfamiliar with data quality and its value to the company.

### 8.3 Critical Success Factors

You too can apply what you have learned here to further high data quality within your organization. Let me share some critical success factors necessary to see (“C”) the program to success:

- **Commitment:** Achieving data quality is possible but it takes time, energy, expertise, and money. There are many choices for using those resources. Like any start-up company, it takes dedication to see the work through and not give up before results are seen.
- **Communication:** Appropriate communication (with an emphasis on appropriate) must take place throughout. Make time for it. If you are fortunate to have management support now, continue to build on it. Don’t take it for granted. Find out how much they want to know, how often, and in what kind of format. Keep them suitably engaged. Expand your communications as needed to other audiences, such as individual contributors (and *their* managers) whose expertise and participation are required.
- **Collaboration, Coordination, and Cooperation:** Data quality will only be achieved by working together. Business is conducted in silos, but data flows between those silos. So working with data requires horizontal thinking. There are many people and teams that will have to cooperate and the data quality team needs to help those relationships. Keep engaged with the business priorities; adjust and align your data quality activities accordingly. Naturally incorporate data quality activities and the data quality message into what is already going on at your company.
- **Change:** Everything we do in data quality seems to trigger change. From adding responsibilities to an already busy role to transforming processes to working with people you haven’t worked with before. Recognize that much of what will be asked will make people uncomfortable. Learn how to manage the human aspect of change. Take your culture and environment into account. Ensure management is ready to defend and enforce change when necessary.
- **Courage:** We don’t often talk about courage in the data quality world, but it does take courage to do something new, propose something different, and lead the way. It takes courage to move out of your comfort zone, stretch yourself, and innovate. It will take courage from the data quality team and from management who will have to defend and enforce change until successful results can speak for themselves.

Some people succeed because they are destined to;  
Most people succeed because they are determined to.

—Anon.

**Acknowledgments** Thanks to those from Company A and Company B—real companies with real people who shared their experiences while choosing to remain anonymous. Thanks to Lisa Jones for her editorial help and Lori Silverman for her content review and suggestions. Special thanks to Shazia Sadiq for her vision for this book and inviting me to contribute.

## References

1. International Association for Information and Data Quality (IAIDQ) (2011) IQCP information quality certified professional. <http://iaidq.org/iqcp/iqcp.shtml> (Accessed 7 Apr 2012)
2. McGilvray D (2008) Executing data quality projects: ten steps to quality data and trusted information™. Morgan Kaufmann, Burlington
3. Mosley M, Brackett M, Earley S (eds) (2010) DAMA guide to the data management body of knowledge (DAMA-DMBOK). Technics Publications, LLC, Bradley Beach
4. Mullins D, Bulkoski R, Zellner A (2012) The war for data talent. Heidrick & Struggles. [http://www.heidrick.com/PublicationsReports/PublicationsReports/HS\\_WarDataTalent.pdf](http://www.heidrick.com/PublicationsReports/PublicationsReports/HS_WarDataTalent.pdf) (Accessed 20 June 2012)
5. Pierce E, Yonke CL, Lintag A (2009) 2009 Information/Data Quality Salary and Job Satisfaction Report: Understanding the Compensation and Outlook of Information/Data Quality Professionals. International Association for Information and Data Quality (IAIDQ). <http://www.iaidq.org/publications/pierce-2009-07.shtml> (Accessed 18 June 2012)
6. Project Management Institute, Inc. (2007) Lexicon of Project Management Terms. <http://www.pmi.org/PMBOK-Guide-and-Standards/PMI-lexicon.aspx> (Accessed 12 June 2012)
7. Redman TC (2008) Data driven: profiting from your most important business asset. Harvard Business School Press, Boston

# Cost and Value Management for Data Quality

Mouzhi Ge and Markus Helfert

**Abstract** The cost and value of data quality have been discussed in numerous articles; however, suitable and rigor cost measures and approaches to estimate the value are rare and indeed difficult to develop. At the same time, as a critical concern to the success of organizations, the cost and value of data quality become important. Numerous business initiatives have been delayed or even cancelled, citing poor-quality data as the main concern. Previous research and practice have indicated that understanding the cost and value of data quality is a critical step to the success of information systems. This chapter provides an overview of cost and value issues related to data quality. This includes data quality cost and value identification, classification, taxonomy, and evaluation framework, as well as analysis model. Furthermore, this chapter provides a guideline for cost and value analysis related to data quality.

## 1 Introduction

In an age characterized by information, data quality is a critical concern for a wide range of organizations. More than ever before, many organizations focus their main business on the provision of valuable information. In order to assure that such information is of a high quality, organizations face considerable challenges of controlling and managing data quality. Data quality management has become an essential component in organizational management e.g., [1–4].

---

M. Ge

Universität der Bundeswehr Munich, Neubiberg (Munich), Germany  
e-mail: [mouzhi.ge@unibw.de](mailto:mouzhi.ge@unibw.de)

M. Helfert (✉)

Dublin City University, Dublin, Ireland  
e-mail: [markus.helfert@computing.dcu.ie](mailto:markus.helfert@computing.dcu.ie)

A substantial body of literature has investigated data quality problems in enterprises. Reports indicate a number of data quality issues in enterprises. It is reported that at least 60 % of enterprises suffer from data quality problems [5], it is also estimated that typically 10–30 % of data in organizational databases are inaccurate [6, 7], an industrial data error rate of 75 % can be found [7], 70 % of manufacturing orders are assessed as of poor data quality [8], 40 % of data in a credit-risk management database was found to be incomplete [9], and between 50 % and 80 % of criminal records are estimated to be inaccurate, incomplete, and ambiguous [10]. Although over the last years some improvements have been made, data quality problems are still pervasive in most enterprises.

Case studies concerning data quality problems are frequently documented in recent years and relate to a broad range of domains. Data quality issues may not only cause errors in business operations but also potentially impact society and wider aspects [64]. For example, in 1986 NASA lost the space shuttle Challenger with seven astronauts onboard. The Presidential Commission investigated the Challenger accident and found that NASA's decision-making process was based on incomplete and misleading data. Just 2 years later the US Navy Cruiser USS Vincennes shot accidentally an Iranian commercial aircraft with 290 passengers onboard. Officials who investigated the Vincennes accident admitted that poor-quality information was a major factor [11]. Yet not only in the space and military industries but also in our daily life, certain data quality problems are found to be severe; for instance, Pirani [12] reported that one piece of wrong biopsy information caused a patient's death in an Australian hospital. Examples such as these illustrate cases in which poor data quality have impact and significant costs and may lead to irreversible damages.

Data quality problems in organizations are often associated with high costs. Olson [13] conducted a survey in 599 companies and found that poor data quality management costs these companies more than \$1.4 billion every year. Also, Huang et al. [14] reported that a telecommunication company lost \$3 million because of data quality problems in customer bills. Moreover, it is estimated that poor data quality results in 8–12 % loss of revenue in a typical enterprise and 40–60 % of expense in service organizations [15]. All the surveys and case studies demonstrate that data quality problems are pervasive, costly, and even disastrous.

Over the last two decades researchers have addressed data quality from different perspectives [71]. Particularly as data quality awareness and requirements have been gradually increased, researchers have focused on data quality frameworks [16, 17], data quality dimensions [9, 18], data quality assessment methods [19, 20, 66], and total data quality management [14, 17]. Also, literature provides us with numerous case studies, investigating data quality issues in practice. Among the topics, managing the cost and benefit of data quality is a crucial aspect concerning the motivation of data quality research [21] and direct interests from industry [22]. Examples from practice and illustrations of implications of poor data quality can be found, for example, in [23, 24].

The data quality literature has consistently emphasized the need for alignment of data quality management with organizational goals and strategies and its impact on organizational performance. One of the key challenges and subject of continuous

debate focuses on how to measure and manage the cost and value of the data quality as well as how to increase and maintain the value that information created to organizations. It has been a concern to researchers, practitioners, and top management for many years that the issues in measuring less tangible cost and value of data quality need to be addressed.

This chapter aims to provide an overview of the cost, benefit, and value of data quality<sup>1</sup> management, which includes cost and value measurement, taxonomy, framework, and analysis in practice. Furthermore, we also provide a guideline for data quality cost and value analysis. Since information creates cost and value not only in financial terms but also in terms of operational and strategic advantages, we believe that this chapter can contribute to the knowledge of successfully building organizational information systems and provide insights for business operations such as decision-making, strategy development, or optimizing supply chains.

## 2 Data Quality Cost

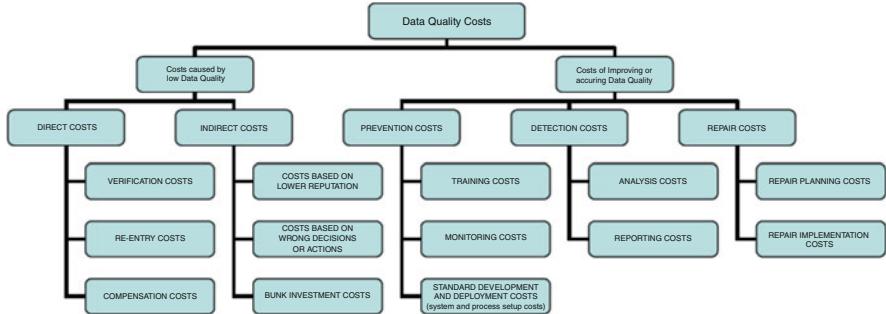
In order to analyze the cost of data quality, it is important to firstly define the concept of data quality cost. In this chapter, we define data quality cost in an enterprise as *financial impacts caused by data quality problems and resources required to achieve a required level of data quality*. It has been found that cost analysis plays an important role in data quality management [65].

### 2.1 Taxonomy of Data Quality Costs

In order to identify the data quality costs, we firstly adopted a frequently cited data quality cost taxonomy developed in our earlier research [21]. A similar and wider discussion can be found in [2]. In their work they discuss the cost and benefit of data quality and introduce and compare various classifications of data quality costs/benefits. Haug et al. [25] review recent contribution to data quality costs and describe impacts of poor-quality data. They also discuss data maintenance efforts and costs inflicted by poor-quality data. Following the above definition, we proposed an extended taxonomy that contains two types of major costs, (1) costs caused by low data quality and (2) costs of improving and assuring data duality. The costs caused by low data quality are divided into direct and indirect costs. Direct costs are those negative monetary effects that are raised directly out of low data quality, namely, the costs of verifying data because it is of questionable credibility, the costs of reentering data because it is inaccurate or incomplete, and the costs of compensation for damages to business based on the low-quality data. On the other hand, indirect costs are those negative monetary effects that arise, through

---

<sup>1</sup>Data quality and information quality are used interchangeably in this chapter.



**Fig. 1** Taxonomy for data quality costs [21]

intermediate effects, from low-quality data. Such indirect costs are loss of a price premium because of a deteriorating reputation, the costs incurred by suboptimal decisions based on poor-quality data, or the investment costs that have to be written off because of low-quality data.

In terms of the costs resulting from improving data quality, Eppler and Helfert [21] distinguished among prevention, detection, and repair costs. Prevention costs refer to the costs that are used to prevent any possible effects caused by data quality problems, for example, costs of training data quality staff or costs of developing data quality standard. Detection costs are the costs related to data quality analysis and profiling. These costs occurred when analyzing and reporting the source of the data quality problems. Examples of repair costs are costs for data quality improvement activities such as repair planning costs and implementation costs (Fig. 1).

Another prerequisite of identifying data quality costs is to classify data quality problems. Many researchers have contributed work on the identification of data quality problems. Garvin [26] pointed out three types of data quality problems: biased information, outdated information, and massaged information. Biased information means the content of the information is inaccurate or distorted in the process of transformation. Outdated information is the information that is not sufficiently up to date for the task. Massaged information refers to different representations of the same information. Lesca and Lesca [27] classified data quality problems into the product and process views. The product view focuses on the deficiencies of the information itself [70], such as incompleteness and inconsistency, whilst the process view concentrates on the problems that are caused in the information manufacturing and distribution process.

Based on the literature review of data quality problems, we propose to classify data quality problems by a two-by-two conceptual model as indicated from [10]. The columns capture data quality problems from both an information perspective and a user perspective, and the rows capture data quality problems as context-independent and context-dependent. Using this model, we classify typical information quality problems in Table 1. This table provides a comprehensive overview of data quality problems. The relevant research contributions we summarized are

**Table 1** Classification of data quality problems

	<b>Information perspective</b>	<b>User perspective</b>
Context-independent	Spelling error [18, 28]	The information is inaccessible [14]
	Missing data [18, 28]	The information is insecure [14]
	Duplicate data [18, 28]	The information is hardly retrievable [14]
	Incorrect value [18, 26, 28]	The information is difficult to aggregate [14]
	Inconsistent data format [18, 26, 28]	Errors in the information transformation [14]
	Outdated data [18, 26, 28]	
	Incomplete data format [14, 28]	
	Syntax violation [28]	
	Unique value violation [29]	
	Violation of integrity constraints [28]	
	Text formatting [26, 28]	
	Violation of domain constraint [27, 28]	The information is not based on fact [14, 26]
	Violation of organization's business rules [27, 28]	The information is of doubtful credibility [14]
	Violation of company and government regulations [27, 28]	The information is irrelevant to the work [14]
Context-dependent	Violation of constraints provided by the database administrator [28]	The information consists of inconsistent meanings [14, 26, 28]
		The information is hard to manipulate [14, 18]
		The information is hard to understand [14, 18]

from [14, 18, 26–28]. The numbers in the table indicate which data quality problem is mentioned in which literature:

- The information perspective/context-independent quadrant indicates data quality problems in the database. These data quality problems can be applied to any dataset.
- The information perspective/context-dependent quadrant indicates data quality problems that violate the business specifications. These data quality problems can be detected by contextual rules.
- The user perspective/context-independent quadrant indicates data quality problems that may occur when processing the information.
- The user perspective/context-dependent quadrant indicates data quality problems that render information not fit for its intended use by information consumers.

With regards to data quality problems, various methods have been applied to solve the problems. From data itself, data quality problems can be solved through data

**Table 2** Costs resulting from low-quality data [21]

1.	Data maintenance costs
2.	Personnel costs
3.	Data search costs
4.	Data quality assessment costs
5.	Semantic confusion costs
6.	Data re-input costs
7.	Wrong data interpretation costs
8.	Time costs of viewing irrelevant information
9.	Loss of revenue
10.	Cost of losing current customer
11.	Cost of losing potential new customer
12.	Cost of realigning business rules
13.	Cost of complicated data integrity
14.	“Loss of orders” cost
15.	Higher retrieval costs
16.	Higher data administration costs
17.	General waste of money
18.	Cost of system migration and reengineering
19.	Costs in terms of lost opportunity
20.	Costs due to tarnished image
21.	Costs related to invasion of privacy and civil liberties
22.	Costs in terms of personal injury and death of people
23.	Costs because of lawsuits
24.	Process failure costs
25.	Information scrap and rework costs
26.	Lost and missed opportunity costs
27.	Costs due to increased time of delivery
28.	Costs of acceptance testing

cleansing algorithms [13], data mining rules [30], statistical process control [7], or dictionary matching routines [10]. From user perspective, data quality problems usually cannot be resolved by automated processes [21]. Solving these data quality problems may require optimization of resource allocation [31], analysis of business issues [17], reengineering processes [7], or aligning information flow with the corresponding information manufacturing systems [8]. All the data quality improvements are often related to costs or organizational resources such as human or hardware support.

## 2.2 Identifying Data Quality Costs

Based on the review of data quality problems and data quality taxonomy, we further identified different types of costs caused by low-quality data from various journal and conference papers. Continuing our earlier work [21], we present a list of possible data quality costs in Table 2.

**Table 3** Costs of assuring data quality [21]

1.	Information quality assessment or inspection costs [32]
2.	Information quality process improvement and defect prevention costs [33]
3.	Preventing low-quality data [26]
4.	Detecting low-quality data [33]
5.	Repairing low-quality data [33]
6.	Costs of improving data format [34]
7.	Investment costs of improving data infrastructures [33]
8.	Investment costs of improving data processes [33]
9.	Training costs of improving data quality know-how [33]
10.	Management and administrative costs associated with ensuring data quality [33]

Poor data quality can generate other negative impacts such as (1) low internal acceptance (poor data quality can lead to a loss of confidence. If data users already distrust a certain data source or data warehouse, the expected value of data warehouse will be lowered), (2) poor decision-making process (decision-making processes are usually based on data analysis and reports. Poor data quality can lead to incorrect evaluations and inconclusive results such as wrong investment decisions or choosing wrong suppliers), and (3) inadequate support for operational business processes (in the operational business processes, poor data quality can impact customer satisfaction and in turn the company image). All the negative consequences can be considered as an indirect cost resulting from poor data quality.

In addition to the costs caused by low-quality data, there are also a number of costs that are associated with assuring data quality (Table 3).

After summarizing the data quality costs, we classify these costs into different groups based on various criteria such as cost origins, effects of the costs, data quality dimensions, and progression of the costs (Table 4).

In order to reduce the data quality costs, a number of studies have made efforts towards managing data quality [6]. These studies typically merged other domain knowledge into data quality management. The merging domains are, for example, total quality management, information management, and knowledge management. From a total quality perspective, Wang [17] proposed a total data quality management (TDQM) framework and advocated the principle of “manage your information as a product.” This framework consists of four stages, define, measure, analyze, and improve, which are adapted from the Deming cycle. The objective of TDQM is to deliver high-quality information products to information consumers. From an information perspective, Eppler [18] proposed an information-focused framework that considers information integration, validation, contextualization, and activation. The objective of this framework is to structure data quality handling and value-adding activities. Also from a knowledge perspective, following the principle “know-what, know-how, know-why,” Huang et al. [14] proposed a framework that is used to transform high-quality data and information into knowledge. This framework contains three key processes: improve quality of information, make tacit knowledge

**Table 4** Data quality costs based on different criteria [21]

A. Data quality costs by origin and life cycle
● Costs due to incorrect capture or entry (time and effort to identify incorrect entries, repairing wrong entries, informing about capture modifications)
● Costs due to incorrect processing
● Costs due to incorrect distribution or communication
● Costs due to incorrect recapture/reentries
● Costs due to inadequate aggregation (e.g., inconsistent aggregations)
● Costs due to inadequate deletion (e.g., data loss)
B. Data quality costs by effect
● Costs of lost customers for marketing
● Costs of scrap and rework in production
● Costs of identifying bad data in operations
● Costs of reentry at data capture point
● Costs of screening at data use points
● Costs of tracking mistakes
● Costs of processing customer data complaints
C. Data quality costs by dimensions
● Costs due to untimely arrival of information (e.g., missed opportunity)
● Costs due to inaccurate information (correction costs)
● Costs due to inaccessible information (higher information gathering costs)
● Costs due to inconsistent information (higher checking and comparing costs)
● Costs due to unreliable information (checking costs)
D. Data quality costs by progression
● Decreasing data costs
● Marginal data costs
● Increasing data costs
● Fixed data costs
● Variable data costs
● Exponential data costs

explicit, and create organizational knowledge. Following these contribution in relation to data quality management, we propose below a guideline to measure data quality costs and value.

### 3 Data Quality Value

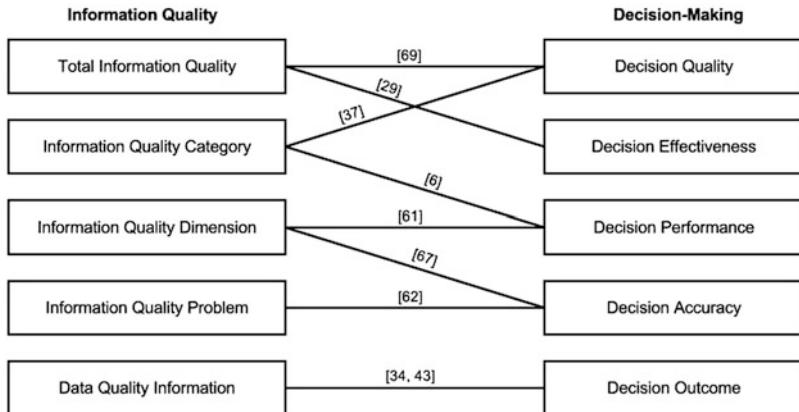
The second element to be considered when evaluating data quality is its impact and value to the business. The aspect of business value in relation to IS has been discussed in numerous papers. For instance, Gustafsson et al. [35] have presented a comprehensive model that aims to explain the business impact with three generic elements: IT, organizational impact, and business value. This model serves as background model for data quality impact. Other related frameworks have been presented in the literature aiming to refine this generic model [36].

The model is supported by strong evidence that data quality has a considerable effect on decision-making in organizations. This section will therefore focus on the data quality value in decision-making. For instance, Keller and Staelin [29] indicate that increasing information quantity impairs decision effectiveness and, in contrast, increasing data quality improves decision effectiveness. Jung et al. conduct a study to explore the impact of representational data quality (which comprises the data quality dimensions' interpretability, easy to understand, and concise and consistent representation) on decision effectiveness in a laboratory experiment with two tasks with different levels of complexity [6]. Furthermore, Ge and Helfert [37] show that the improvement of data quality in the intrinsic category (e.g., accuracy) and the contextual category (e.g., completeness) can enhance decision quality. In Fig. 2, we summarize the key contributions investigating the relationship between data quality and decision-making. Although the terms are different, on closer examination, the primary measurements are often similar.

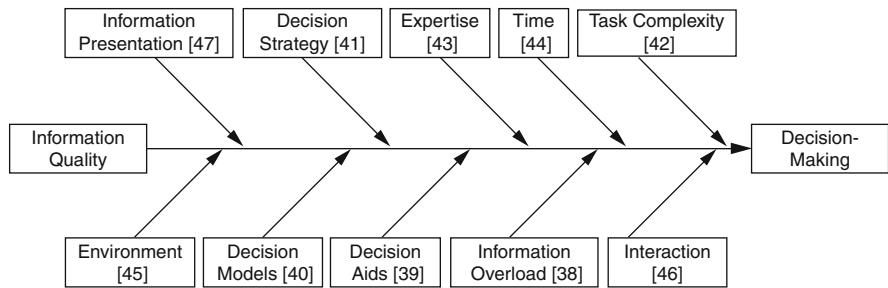
However, when investigating the impact and value of data quality in organizations, there are many factors that may influence the relationship. Besides data quality, other factors may also influence decision quality, such as information overload [38], decision aids [39], decision models [40], decision strategy [41], task complexity [42], expertise [43], decision time [44], decision environment [45], interaction [46], and information presentation [47]. These factors are summarized in Fig. 3.

In order to investigate the business value of data quality, we follow IS business value studies that show how IS impacts on business processes and/or decision-making. A business process can be defined as “a specific ordering of work activities across time and place, with a beginning, an end, and clearly identified inputs and outputs: a structure for action” [33]. Porter and Millar argue that activities that create value consist of a physical and an information-processing component and each value activity uses information [48]. In their integrative model of IS business value, Mooney et al. [49] propose a process framework for assessing the IS business value. They present a typology of processes that subdivides business processes into operational and management processes and argue that IS creates business value as it has automational, informational, and transformational effects on the processes. Similarly, Melville et al. [50] see business processes and business process performance as the key steps that link IS resources and complementary organizational resources to organizational performance. Data can be seen as an important organizational asset as well as resource. Its quality is directly related to business value and organizational performance.

In addition to measuring the effect on business processes, organizational performance has always been of consideration to IS researchers and practitioners, resulting in a plethora of performance-related contributions. Earlier approaches focused, for example, on the economic value of information systems [51]. They were more recently detailed to frameworks for assigning the impact of IS to businesses [49, 50]. These IS-oriented frameworks have resulted in an abundance of recommendations, frameworks, and approaches for performance measurement systems.

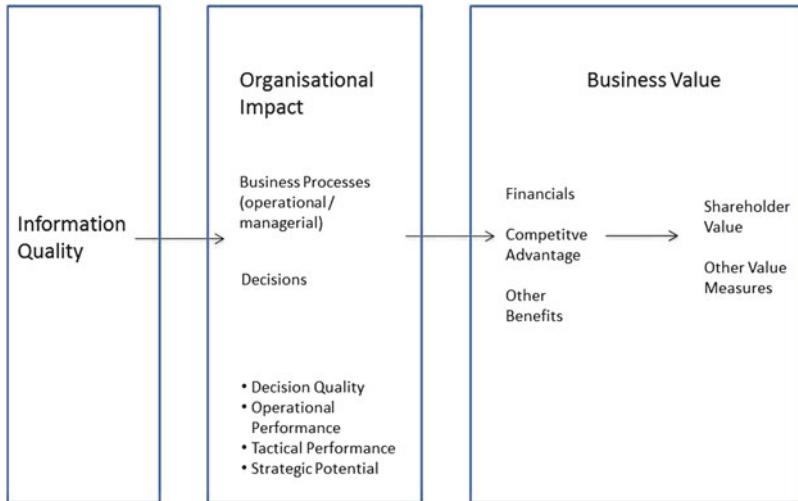


**Fig. 2** Data quality value in decision-making



**Fig. 3** Factors influencing the relationship between data quality and decision-making

It has been recognized that there are two perspectives on value – objective and perceived value – which results in different data quality and value measures and value perceptions for particular stakeholders [63]. To evaluate the value of data quality and develop suitable indicators, we suggest combining the work on business processes and decision quality with the work on performance indicators, developing a framework for analyzing business value of data quality. This is illustrated in Fig. 4. The value propositions of data quality are manifold. It ranges from generating direct business value by providing information of high quality, reducing complexity, improving customer loyalty, improving operational performance, reducing costs, and so on. Due to its critical importance in enterprises, data quality affects many areas of organizational performance and may deliver business value simultaneously to stakeholders.



**Fig. 4** Framework for analyzing business value of data quality

## 4 Cost and Value Model for Data Quality

Based on economic theory, Eppler and Helfert [21] have reviewed the major quality cost models in manufacturing and then provide a basis for linking the existing quality cost theory to data quality costs. One of the first quality cost models was developed by [52]. This model is based on the classification of the quality costs, which contains three main categories: prevention, appraisal, and failure costs. Crosby [32] developed the concept of a process-oriented quality cost model, which determines the total cost of quality by summing costs of achieving quality (costs of conformance) and costs due to lack of quality (costs of nonconformance). Costs of conformance and costs of nonconformance are typically inversely related to each other. For example, more investment in achieving high quality can lead to a decrease of costs due to the lack of quality. This relationship and its effects on the total cost of quality are normally shown as curves related to the quality level and can be described as a quality metric.

In traditional cost and value models, total quality cost has implicated the diminishing returns, which is a minimum prior to achieving the maximum quality. A cost model has a total cost minimum prior to achieving 100 % of quality. However, in manufacturing research has shown that increased attention to defect prevention leads to large reduction in evaluation costs. This experience is reflected in modern total quality management philosophies with a focus on continuous improvement of data quality.

Most of the cost models used in the context of manufacturing and service are well understood and interpreted. For these cost models, there exist some assumptions and realistic cost estimations. In contrast to these models, data quality cost

progression is still limited understood. In addition the actual cost curve progression is highly domain-dependent and extremely difficult to estimate in practice. It largely depends on the corresponding information systems and subjective estimation from information consumers.

Recent work on data quality cost models includes work on risk management, meta-models, and evaluating data quality management approaches concerning its costs and business value. For instance, Even et al. [53] propose an evaluation model for cost-effective data quality management that links the handling of data quality defects to economic outcomes such as utility, cost, and net benefit. The model is demonstrated in a real-world CRM setting [54]. Even and Shankaranarayanan [53] propose utility cost perspectives in data quality management. Batini and Scannapieca [2] discuss the relevance of selected classifications of data quality costs/benefits and their incorporation into and relation to data quality methodologies. Haug et al. [25] propose in their work a clarification of costs of poor-quality data and define, based on a quality cost model, the relation to data quality maintenance effort. Furthermore, Otto [55] proposes a meta-model for simulating data quality management, which can be used for planning of cost-efficient initiatives. Borek et al. [56] relate IS resources from an information quality perspective to business value. Klassen et al. [57] propose a risk-based approach to quantifying the business impact of information quality. Expanding the cost perspective Ryu et al. [3] propose a data quality management maturity model.

In summary, most of the recent work aims to relate cost and value in DQ cost models or provide foundation approaches for estimating both. Both aspects are represented in our guideline below as value and cost indicators. Limited studies are still available on describing typical cost and value progressions in the form of patterns. In our work [21] we argue that we can derive the following observations from the experiences in previous data quality projects:

- Costs caused by low data quality: these costs highly depend on the context and the subjective estimation from data consumers. In practice, the costs caused by low data quality should be expected to decrease with the increasing quality. It might be realistic for many contexts to assume that this cost curve is convex.
- Repair costs: in manufacturing a convex curve of cost is typically assumed. That means repair costs are zero when the quality is at a maximum level. For data quality costs, the predominant role of humans in the data creation process and limited capabilities of automatic data cleansing methods further support this assumption.
- Detection costs: similar as repair costs, detection costs for data quality can also be assumed. There exist a variety of tools for detecting data quality problems, and in some cases data repair and corrections have to be carried out manually. Thus when data quality is at a relatively high level, detection cost is often lower than the repair cost.
- Total costs: total data quality costs are determined by summarizing all the involved costs. Total costs can be considered as a sum of costs caused by low data quality, detection costs, and repair costs. Depending on the particular cost

and value characteristics, an optimum of total data quality cost can be reached at a point before maximum quality. This is the point when costs caused by low data quality and costs for assuring data quality are balanced.

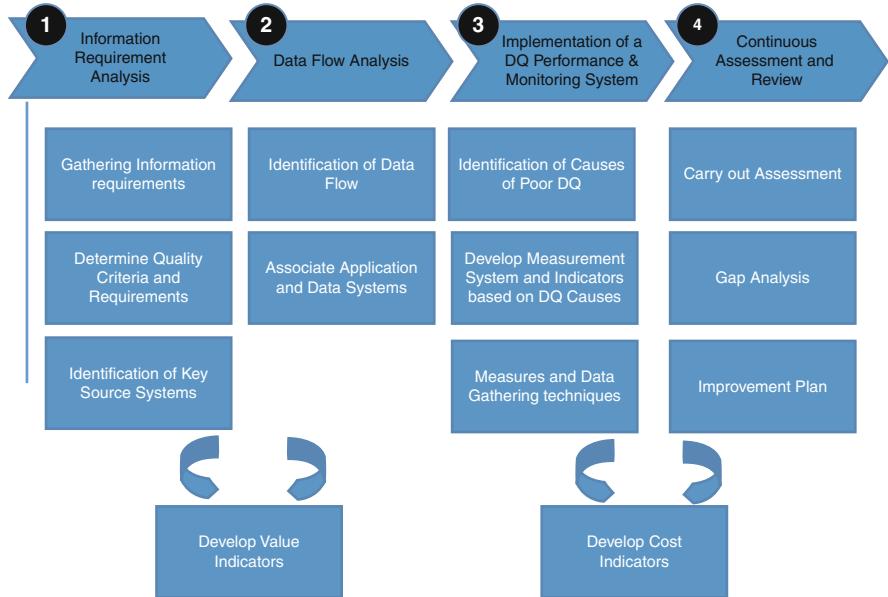
It has been argued that preventing quality problems results in significantly reduced repair and detection costs [58–60]. This is due to the assumption that the sooner a defect is detected or prevented, the more savings can be made during subsequent processes. Although [19] or [7] also described similar observations in data quality improvement projects, considering the limited research in data quality costs, so far it is still unfeasible to quantify the effects of data quality prevention measures on repair and detection costs. However from the experience in manufacturing, software engineering, and data quality projects, we assume that increased prevention costs can result in significant savings of repair and detection costs. This assumption is subject to justify due to the increasing complexity of informational environments, numerous data duplications, and interlinking of data chains. When proposing the data quality cost and value model, we also considered the dependencies and trade-offs of data quality dimensions [52]. Dependencies can be used in the improvement phase to improve its efficiency. In the improvement process, each improvement action can impact on a specific subset of quality dimensions. For example, data cleaning focuses on accuracy and consistency dimensions, data enrichment improves source completeness, and source duplication is for data availability improvement. In order to have a total data quality program, it is necessary to consider more than one action to increase the overall quality level, and thus it is necessary to design the so-called improvement plan. Dependencies among dimensions can be used as drivers for the selection of the improvement actions. Also, we can consider dependencies among dimensions for the definition of the order with which actions should be executed. Due to the fact that if the quality dimension one depends on the quality dimension two, improvements performed on the two would also increase the quality of dimension one.

## 5 Guideline for Cost and Value Analysis

In order to provide a cost and value analysis guideline for practitioners, we have developed a framework in the form of a guideline for cost and value analysis as illustrated in Fig. 5. The framework can help guide enterprises during data quality cost and value evaluation process together with its key elements. We propose to establish value indicators and then subsequently cost indicators. Both can be used to develop suitable improvement measures.

The data quality cost and value framework is divided into four phases, which can be used to evaluate the cost and value in data quality management. We describe the four phases as follows:

- Phase 1 is to identify and analyze the information requirements. This phase includes gathering the information requirements, determining quality criteria,



**Fig. 5** Data quality cost-value evaluation framework

and identifying system resources, which are used to establish the specification for the information manufacturing system.

- Based on phase 1, phase 2 is to describe the information flow and its usages in the business processes. This phase will identify the data flow, related application, and business processes such as decision-making process. Thus, we can derive data quality value indicators and describe the business value contribution of an information product.
- After determining data quality value, phase 3 is to develop assessment measurement and analyze root causes of data quality problems. This phase will identify the data quality problems and estimate the related costs.
- From a continuous data quality improvement view, phase 4 is to develop an improvement plan in the long run, which includes reviewing the data quality assessment and analyzing the gaps in data quality management. Together with phase 3, the last two phases are used to develop cost indicators and estimate the costs in data quality management.

## 6 Summary and Conclusion

This chapter discussed the cost and value measurement in data quality research and practice. Based on the literature review and our earlier work [21], we identified a variety of data quality costs and classified the costs based on different criteria

such as cost origins or cost progressions. This classification enables us to propose a data quality cost taxonomy and framework, which can be used for further cost and benefit analysis. We provided an outline of a practice-oriented data quality cost-value evaluation framework.

We believe that if data quality research is to make significant progress in terms of its acceptance in the business world, the costs associated with low data quality must be more explicit, prominent, and measurable. By incorporating the costs of assuring data quality, our proposed model can help companies to determine an optimal investment in data quality projects and can be used as a guideline for cost and value analysis in data quality.

Over the last decade, a move from solely technical data quality aspects to a combination of multiple contextual factors such as technology, sector, organizational culture, and management style can be observed [19]. It has been argued that data quality management can result in enhanced business value [68]. We view data and information as a crucial asset that creates value and knowledge to organizations and thus need to be managed accordingly for optimization analysis. However the issues of how to make the business impact of data quality more visible and how to manage the cost and benefit in data quality project are still challenging nowadays. In this chapter, we have provided a platform for discussing approaches, models, results, and case studies addressing a broad range of issues related to cost and benefit management in data quality projects. We believe the findings and discussions in this chapter are beneficial for the future research and real-world projects.

## References

1. Helfert M, Herrmann C (2005) Introducing data quality management in data warehousing. In: Wang RY, Madnick SE, Pierce EM, Fisher CW (eds) *Information quality. Advances in management information systems*, vol 1. M.E. Sharpe, Armonk, pp 135–150
2. Batini C (2006) *Monica Scannapieca data quality concepts, methodologies and techniques*. Springer, Heidelberg
3. Ryu K-S, Park J-S, Park J-H (2006) A data quality management maturity model. *ETRI J*, 28(2):191–204. doi:10.4218/etrij.06.0105.0026
4. Otto B, Hüner KM, Österle H (2012) Toward a functional reference model for master data quality management, *Inf Syst e-Bus Manage* 10(3):395–425. ISSN:1617–9846, 09/2012
5. Wand Y, Wang RY (1996) Anchoring data quality dimensions in ontological foundations. *Commun ACM* 39(11):86–95
6. Jung W, Olfman L (2005) An experimental study of the effects of contextual data quality and task complexity on decision performance, *Information Reuse and Integration Conference*, Las Vegas, NV
7. Redman T (1996) *Data quality for the information age*. Artech House, Boston
8. Wang RY, Lee Y, Ziad M (2001) *Data quality*. Kluwer Academic, Norwell
9. Wang RY, Strong DM (1996) Beyond accuracy: what data quality means to data consumers. *J Manage Inf Syst* 12(4):5–34
10. Strong DM, Lee YW, Wang RY (1997) Data quality in context. *Commun ACM* 40(5):103–110
11. Fisher CW, Kingma BR (2001) Criticality of data quality as exemplified in two disasters. *Inf Manage* 39(2):109–116

12. Pirani C (2004) How safe are you hospital? *The Weekend Australia*
13. Olson JE (2003) Data quality: the accuracy dimension. Morgan Kaufmann, San Francisco
14. Huang KT, Lee Y, Wang RY (1999) Quality information and knowledge management. Prentice Hall, Upper Saddle River
15. Redman T (1998) The impact of poor data quality on the typical enterprise. *Commun ACM* 41(2):79–82
16. Ballou DP, Pazer HL (1985) Modeling data and process quality in multi-input, multi-output information systems. *Manage Sci* 31(2):150–162
17. Wang RY (1998) A product perspective on total data quality management. *Commun ACM* 41(2):58–65
18. Eppler M (2006) Managing information quality, 2nd edn. Springer, Berlin
19. English LP (1999) Improving data warehouse and business information quality: methods for reducing costs and increasing profits. Wiley, New York
20. Pipino L, Lee Y, Wang RY (2002) Data quality assessment. *Commun ACM* 45(4):211–218
21. Eppler M, Helfert M (2004) A classification and analysis of data quality costs. In: 9th MIT international conference on information quality, November 5–6, 2004, Boston, MA
22. Fisher T (2009) The data asset: how smart companies govern their data for business success. Wiley, New Jersey, p 220
23. Anderson T (2005) The penalties of poor data, 2005. Whitepaper published by GoImmedia.com and the Data Warehousing Institute dw-institute.com. <http://www.goimmedia.com/ArticlesWhitepapers/ThePenaltiesofPoorData.aspx>
24. O'Brien T (2009) Quality counts - why poor data quality can cost you money and also have legal implications. Governance and Compliance. <http://www.icsaglobal.com/governance-and-compliance/features/jan-2011-quality-counts#UKyvx-Tgm-Q>
25. Haug A, Zachariassen F, van Liempd D (2011) The costs of poor data quality, *J Ind Eng Manage* 4(2):168–193
26. Garvin DA (1988) Managing quality. Free Press, New York
27. Lesca H, Lesca E (1995) Gestion de l'information: qualité de l'information et performances de l'entreprise. LITEC, Les essentiels de la gestion
28. Oliveira P, Rodrigues F, Henriques P (2005) A formal definition of data quality problems. In: 11th international conference on information quality, Boston, MA
29. Keller KL, Staelin R (1987), Effects of quality and quantity of information on decision effectiveness. *J Consum Res* 14(2):200–213
30. Savchenko S (2003) Automating objective data quality assessment. In: 8th international conference on information quality, Boston, MA
31. Ballou DP, Tayi GK (1989) Methodology for allocating resources for data quality enhancement. *Commun ACM* 32(3):320–329
32. Crosby PB (1979) Quality is free. McGraw-Hill, New York
33. Davenport TH (1993) Process innovation: reengineering work through information technology. Harvard Business Press, Boston
34. Chengalur-Smith IN, Ballou DP, Pazer HL (1999) The impact of data quality information on decision making: an exploratory analysis. *IEEE Trans Knowl Data Eng* 11(6):853–864
35. Gustafsson P, et al (2008) Quantifying IT impacts on organizational structure and business value with Extended Influence Diagrams. In: Stirna J, Persson A (eds) The practice of enterprise modeling. Lecture Notes in Business Information Processing, vol 15, pp 138–152
36. Borek A, Helfert M, Ge M, Parlakad AK (2011) An information oriented framework for relating IS resources and business value. In: Proceedings of the international conference on enterprise information systems, Beijing
37. Ge M, Helfert M (2008) Effects of information quality on inventory management. *Int J Inf Qual* 2(2):176–191
38. Eppler MJ, Mengis J (2003) The concept of information overload: a review of literature from organization science, accounting, marketing, MIS, and related disciplines. *Inf Soc Int J* 20(5):1–20
39. Neumann S, Hadass M (1980) DSS and strategic decisions. *Calif Manage Rev* 22(2):77–84

40. Janis I, Mann L (1977) Decision making: a psychological analysis of conflict, choice and commitments, The Free Press, New York
41. Payne JW, Bettman JR, Johnson EJ (1988) Adaptive strategy selection in decision making. *J Exp Psychol Learn Mem Cogn* 14(3):534–552
42. Campbell DJ (1988) Task complexity: a review and analysis. *Acad Manage Rev* 13(1):40–52
43. Fisher CW, Chengalur-Smith I, Ballou DP (2003) The impact of experience and time on the use of data quality information in decision making. *Inf Syst Res* 14(2):170–188
44. Svenson O, Edland A, Slovic P (1990) Choices and judgments of incompletely described decision alternatives under time pressure. *Acta Psychol* 75:153–169
45. Shankaranarayanan G, Ziad M, Wang RY (2003) Managing data quality in dynamic decision environments: an information product approach. *J Data Manage* 14(4):14–32
46. Burleson BR, Levine BJ, Samter W (1984) Decision making procedure and decision quality. *Hum Commun Res* 10:557–574
47. Remus W (1984) An investigation of the impact of graphical and tabular data presentations on decision making. *Manage Sci* 30(5):533–542
48. Porter ME, Millar V (1985) How information gives you competitive advantage. *Harv Bus Rev* 63(4):149–160
49. Mooney JG, Gurbaxani V, Kraemer KL (1996) A process oriented framework for assessing the business value of information technology. *ACM SIGMIS Database* 27(2):68–81
50. Melville N, Kraemer K, Gurbaxani V (2004) Information Technology and Organizational Performance: An Integrative Model of IT Business Value. *MIS Q* 28(2):283–322
51. Van Wegen B, De Hoog R (1996) Measuring the economic value of information systems. *J Inf Technol* 11(3):247–260
52. Feigenbaum AV (1956) Total quality control. *Harv Bus Rev* 34:93–101
53. Even A, Shankaranarayanan G (2009) Utility cost perspectives in data quality management. *J Comput Inf Syst* 50(2):127–135
54. Even A, Shankaranarayanan G, Berger PD (2010) Evaluating a model for cost-effective data quality management in a real-world CRM setting. *Decis Support Syst* 50(1):152–163
55. Otto B, Hüner KM (2009) A meta-model for data quality management simulation. In: 14th international conference on information quality, Potsdam. <http://works.bepress.com/boris-otto/11>
56. Borek A, Helfert M, Ge M, Parlikad AK (2011) IS resources and business value: operationalisation of an information oriented framework. In: Enterprise information systems - revised selected papers. Lecture Notes in Business Information Processing. Springer, Heidelberg
57. Klassen V, Borek A, Kern R, Parlikad AK (2012) Quantifying the business impact of information quality - a risk based approach. In: Proceedings of the 20th European Conference on Information Systems (ECIS), Barcelona
58. Cappiello C, Francalanci C (2002) Considerations about costs deriving from a poor data quality. DaQuinCIS Project Report, December 2002
59. Gryna FM (1998) Quality and costs. In: Juran JM, Godfrey AB (eds) Juran's quality handbook, 5th edn. McGraw-Hill, New York
60. Segev A, Wang R (2001) Data quality challenges in enabling eBusiness transformation. In: Pierce E, Katz-Haas R (eds) Proceedings of the sixth MIT information quality conference, 2001, Boston, MA, pp 83–91
61. Ahituv N, Igbaria M, Stella A (1998) The effects of time pressure and completeness of information on decision making. *J Manage Inf Syst* 15(2):153–172
62. Ballou DP, Pazer HL (1990) A framework for the analysis of error conjunctive, multi-criteria, satisfying decision processes. *Decis Sci* 21(4):752–770
63. Fehrenbacher DD, Helfert M (2012) Contextual factors influencing perceived importance and trade-offs of information quality. *Commun Assoc Inf Syst* 30:Article 8
64. Klein BD, Goodhue DL, Davis GB (1997) Can humans detect errors in data? Impact of base rates, incentives, and goals. *MIS Q* 21(2):169–194
65. Kim W, Choi B (2003) Towards quantifying data quality costs. *J Object Technol* 2(4):69–76

66. Lee Y, Strong DM, Kahn B, Wang RY (2002) AIMQ: a methodology for information quality assessment. *Inf Manage* 40(2):133–146
67. Letzring TD, Wells SM, Funder DC (2006) Information quantity and quality affect the realistic accuracy of personality judgment. *J Personal Soc Psychol* 91(1):111–123
68. Neus A (2001) Managing information quality in virtual communities of practice: lessons learned from a decade's experience with exploding Internet communication. In: Pierce E, Katz-Haas R (eds) *Proceedings of the sixth MIT information quality conference, 2001*, Boston, MA, pp 119–131
69. Raghunathan S (1999) Impact of information quality and decision-maker quality on decision quality: a theoretical model and simulation analysis. *Decis Support Syst* 26(4):275–286
70. Verykios VS, Elfeky MG, Elmagarmid AK, Cochinwala M, Dalal S (2000) On the accuracy and completeness of the record matching process. In: Klein BD, Rossin DF (eds) *Proceedings of the 5th international conference on information quality*, MIT, Boston
71. Wang RY, Storey VC, Firth CP (1995) A framework for analysis of data quality research. *IEEE Trans Knowl Data Eng* 7(4):623–640

# On the Evolution of Data Governance in Firms: The Case of Johnson & Johnson Consumer Products North America

Boris Otto

**Abstract** Data Governance defines decision-making rights for company-wide use of data. The topic has received increased attention both in the scientific and in the practitioners' community, as the quality of data meanwhile is increasingly being considered a key prerequisite for companies for being able to meet a number of strategic business requirements, such as compliance with a growing number of legal provisions or pursuit of a customer-centric business model. While first results can be found in literature addressing Data Governance arrangements, no studies have been published so far investigating the evolution of Data Governance over time. Drawing on theory about organizational capabilities, the chapter assumes that Data Governance can be considered a dynamic capability and that the concept of capability lifecycles can be applied. A single-case study conducted at Johnson & Johnson Consumer Products, North America, is presented to explore the research question as to how Data Governance effectiveness can be measured as the ratio of the number of preventive data quality management (DQM) measures to the total number of DQM measures in order to trace the evolution of Data Governance over time. The findings suggest that Data Governance can in fact be seen as a dynamic capability and that its effectiveness evolves according to a lifecycle curve. Furthermore, the chapter discusses a maturity model which can be used as an instrument to manage and monitor this evolution.

## 1 Introduction

*Data Governance* defines decision-making rights and responsibilities with regard to the use and management of enterprise data [1, 2]. Over the last years Data Governance has received increased attention within the practitioners' community.

---

B. Otto (✉)

University of St. Gallen, St. Gallen, Switzerland

e-mail: [boris.otto@unisg.ch](mailto:boris.otto@unisg.ch)

On June 30, 2005, for example, IBM announced the foundation of the “Data Governance Council,” which then included more than twenty large enterprises from various industries [3]. The Americas’ SAP Users Group, to mention another example, is running a special interest group on Data Governance, which aims to promote the exchange of experiences and “best practices.”<sup>1</sup> Enterprises have begun to consider Data Governance a promising approach to address new requirements on data quality as a consequence of market dynamics.

One example can be found in the French consumer packaged goods (CPG) industry. For every single item shipped to stores CPG manufacturers must provide information about the “carbon footprint” associated not only with their own production and distribution activities but also with their suppliers [4]. This requirement can only be met by applying clear standards along the entire supply chain, defining what data must be available in what quality in which business process at a specific point in time. Another example relates to changing business models in the telecommunications industry. Whereas in the past the core business object literally was the “landline plug in the wall,” telecommunications service providers today must be able to center around each single customer accurate, complete, and timely data on contracts, billing status, network infrastructure availability, and services provided. Consulting company Deloitte summarizes this development in the telecommunications industry as follows: “Data ascends from the basement to the boardroom” [5].

The information systems (IS) research community has taken up on the issue only recently. Weber et al. [1] see Data Governance as aiming at providing a structural framework for decision-making rights and responsibilities with regard to the use of data. Otto [6] has provided a taxonomy for Data Governance organization. And Pierce et al. [7] have investigated the current status of adoption of Data Governance in enterprises.

Major parts of the literature on Data Governance, though, take a “one-off” perspective on the topic and assume, at least implicitly, stability of Data Governance arrangements over time. This conceptualization is too limited and does not live to the fact that Data Governance is a response to market dynamics and, thus, has a lifecycle during which it evolves. It is this gap in literature that has motivated the work presented in this chapter.

## 2 Fundamental Concepts

### 2.1 *Data and Data Quality*

Although numerous scientific studies have dealt with the differentiation between *data* and *information*, a clear, unambiguous, and widely accepted understanding

---

<sup>1</sup>See <http://www.asug.com/Communities/ASUGSpecialInterestGroups.aspx> for details.

of the two terms does not exist [8, 9]. One strand of research sees information as knowledge exchanged during human communication, whereas another takes an information processing lens according to which pieces of data are the building blocks of information [10]. The aim of the chapter is not to take part in that discussion but to follow one specific definition, which is to view information as processed data [11–13].

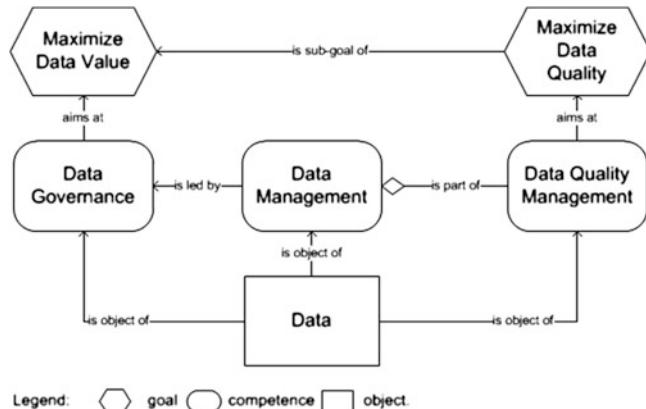
*Data quality* typically is determined by data's "fitness for use," i.e., the capability of data to meet certain requirements defined by the user in order to accomplish a certain goal in a given context [14, 15]. Thus, data quality seems to be in the "eye of the beholder" (i.e., the user). A number of variables—so-called data quality dimensions—are commonly used to determine data quality. Examples of such dimensions are completeness, consistency, accurateness, relevance, or timeliness [16].

## 2.2 ***Data Governance, Data Management, and Data Quality Management***

A standard definition of the term Data Governance can be found neither in the research community nor in the practitioners' community dealing with information systems. However, proposals to define the term have in common that Data Governance refers to the allocation of decision-making rights and related duties in the management and use of enterprise data. According to Weber et al. [1], for example, Data Governance specifies a structural framework for decision-making rights and responsibilities regarding the use of data in an enterprise. Khatri and Brown [2] see Data Governance referring to the assignment of decision-making rights with regard to an enterprise's "data assets."

Data Governance aims at maximizing the value of data assets in enterprises. Viewing data as an asset goes back to the 1980s, when methods and knowledge regarding the management of physical goods for the first time were transferred to the domain of managing immaterial goods, like information and data [17]. Generally, data only has a value if it is being used. As mentioned above, data's "fitness for use" is what Wang [13] considers as data quality. Poor data quality reduces the value of data assets in an enterprise if their utility is low ([18], p. 80). Thus, enterprises are anxious to maximize data quality.

Maximizing data quality is the aim of DQM. DAMA International ([19], p. 20) defines data quality management as the total of activities for "measuring, evaluating, improving, and ensuring data's fitness for use." DQM thereby is a sub-function of data management, which comprises planning, controlling, and provisioning of data assets ([19], p. 4). One can distinguish between preventive and reactive DQM measures [20–22]. Preventive measures are, for example, data quality checks before data entry, data architecture and data lifecycle management, or training of employees. Reactive measures include data correction and data cleansing after a data defect has occurred.



**Fig. 1** Data Governance, data management, and data quality management

Considering the above, Data Governance provides a decision-making framework for *data management*. The relationship between “governance” and “management” follows the ISO definition in the field of information technology (IT) in general [23]. In this sense, Data Governance is about determining which decisions are to be made regarding the use of data and who is involved in decision-making, whereas data management is about making decisions and implementing them. Figure 1 illustrates the relationship of these fundamental concepts.

Apart from that, this chapter defines *Data Governance effectiveness* as the ratio of the number of preventive DQM measures to the total number of DQM measures taken by a company [20, 21]. The rationale behind this definition is as follows: The more DQM is dominated by preventive measures—data quality gates upon data entry, clear responsibilities for definition, and use of data, for example—the higher the effectiveness of Data Governance.

### 3 Related Work

#### 3.1 Data Governance

Apart from very few studies relating to specific domains such as research in healthcare, the concept of Data Governance is basically discussed in the IS and the computer science communities (cf. [24]). Most studies on Data Governance in IS research are characterized by taking a certain theoretical perspective on the topic (e.g., using contingency theory to examine the organizational fit of Data Governance). Furthermore, case studies and reports on practical experiences can be found. Apart from that, many papers refer to Data Governance when explaining what has motivated their research. Kooper et al. [25] also elaborate on the differences between Data Governance and other governance approaches, mainly

focusing on IS governance. No contribution from the IS research community could be found which addresses the dynamic aspects of Data Governance. Moreover, the theoretical discourse in this discipline is limited in terms of the above-mentioned papers adopting an organization theoretical lens.

The computer science community uses Data Governance mainly as a motivation for the design of new models, algorithms, and systems. However, all of these papers fall short in giving a clear definition of Data Governance or addressing the challenges of establishing it.

Table 1 summarizes the review of literature on Data Governance.

### 3.2 *Organizational Capabilities*

The concept of organizational capabilities is an extension of the *resource-based view (RBV)* of the firm. RBV basically forms a theoretical framework for explaining how competitive advantage of enterprises can be achieved [47, 48]. A resource can be defined as an “asset or input to production (tangible or intangible) that an organization owns, controls, or has access to on a semi-permanent basis” ([49], p. 999). If resources meet the *VRIN criteria* (i.e., being *valuable*, *rare*, *inimitable*, and *non-substitutable*), they are likely to yield competitive advantage [50]. Resources are closely related to organizational capabilities, which are defined as the “ability of an organization to perform a coordinated set of tasks, utilizing organizational resources, for the purpose of achieving a particular end result” ([49], p. 999).

Furthermore, one can distinguish *dynamic capabilities* from other organizational capabilities [51] by taking an organizational change perspective. Dynamic capabilities describe an enterprise’s ability to address a changing market environment by integrating, reconfiguring, gaining, and releasing resources [52]. Dynamic capabilities can take many forms, ranging from combination of heterogeneous skills in product development to acquisition of new routines. Moreover, there is the notion of a hierarchical order of capabilities depending on their contribution to the competitive position of an enterprise. Wang and Ahmed [53] give the example of quality control on the one hand and total quality management (TQM) on the other, outlining that TQM is a capability of higher order compared to quality control, as it is more difficult to be imitated by contenders.

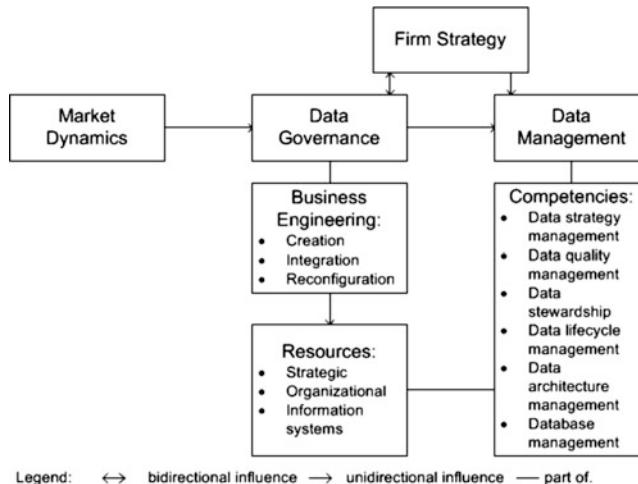
Additionally, Helfat and Peteraf [49] have introduced the notion of the *capability lifecycle*, describing the evolution of capabilities, starting from an initial stage to a development stage and continuously moving on to a stage of maturity.

### 3.3 *Data Governance as a Dynamic Capability*

Figure 2 shows the conceptual framework which guides the course of the argumentation. The framework draws on the theoretical foundations of organizational

**Table 1** Data Governance literature review

Scientific discipline	Main focus	Papers
Information systems (IS)	Application of organizational theories (e.g., contingency theory, organizational design theory) to Data Governance	[1, 2, 6, 26–29]
	Case studies	[30, 31]
	Data governance as a rationale (e.g., for identity management, information security management) or enabler (e.g., for risk management)	[32–36]
	Differentiation of Data Governance from related concepts	[25]
	Practical experiences with Data Governance	[7, 37]
	Data Governance as a rationale to design algorithms, models, systems, etc.	[38–43]
	Frameworks for Data Governance	[44]
Others	Data Governance in e-government, healthcare, etc.	[45, 46]

**Fig. 2** Data Governance as a dynamic capability

capabilities, arguing that both Data Governance and data management can be interpreted as organizational capabilities. Furthermore, this chapter argues that Data Governance forms a dynamic capability, as it is built by enterprises as a response to a changing market environment, posing new requirements on the nature enterprises

manage data and data quality. Following the definition of dynamic capabilities, Data Governance is an organization's ability to rearrange its resources in order to maintain and/or improve its competitive position in times of permanent market change (cf. [49, 52, 54]).

*Business Engineering* as a method-driven and model-oriented approach for transformation [55, 56] is a way to make use of such rearrangement abilities. Concrete Business Engineering activities comprise creation of new resources and integration and configuration of existing resources. Again following the understanding of Business Engineering, resources can be categorized as strategic, organizational, and related to information systems. Strategic resources to be rearranged by Data Governance are goal systems and corporate standards, organizational resources comprise people and processes, and information systems resources are specific master data management (MDM) systems, for example.

Following the argumentation introduced by Wang and Ahmed [53], dynamic capabilities are “higher-order” organizational capabilities and are often related to “lower-order” organizational capabilities. Thus, Data Governance is related to data management competencies. As described above, a variety of normative framework exists for data management. For the following course of the investigation, the conceptual framework adapts the six competencies introduced by [57, 58]. These competencies are “data strategy management,” “data quality management,” “data stewardship,” “data lifecycle management,” “data architecture management,” and “database management.”

## 4 Goal and Approach of the Study

### 4.1 Goal

The chapter wants to shed light on the dynamic aspects of Data Governance. It aims at responding to the question as to how Data Governance effectiveness evolves over time. As not much knowledge is available today about Data Governance effectiveness and its evolution, the chapter takes an explorative approach, presenting the results of a single-case study [59, 60] conducted at Johnson & Johnson’s Consumer Products division in North America<sup>2</sup>.

The chapter makes two important contributions to the scientific body of knowledge. First, it is among the first to address the evolution of Data Governance and its effectiveness over time. Second, it helps advance the understanding of the evolution of organizational capabilities in general. The chapter thereby responds to a need

---

<sup>2</sup>In the following referred to as Johnson & Johnson

in research of being “[s]hort of definitive empirical evidence regarding the exact shape of the capability lifecycle during the founding and development stages” ([49], p. 1004).

## 4.2 Approach

The chapter uses case study research to investigate the evolution of Data Governance. Case study research is adequate if the phenomenon under investigation cannot or should not be isolated from its context and if it is still relatively unexplored [59, 61]. The research design follows the five guiding points proposed by Yin ([59], pp. 21–28).

The research question (1st guiding point) is derived from the research problem, namely, limited knowledge about the evolution of Data Governance in firms over time. Therefore, the central research question is as follows: How does Data Governance effectiveness evolve in firms over time?

Due to the limited amount of scientific knowledge with regard to this question, the study is of exploratory nature. Yin [59] concedes that in exploratory case studies it is unlikely to base one’s research on clear propositions (2nd guiding point). However, he stipulates that case study research should have some purpose (*ibid.*, [59] p. 22) and that it should follow a set of criteria guiding the investigation. The chapter uses a conceptual framework (see Fig. 2) as a guiding scheme for the investigation. The unit of analysis (3rd guiding point) sets the boundaries for the case with regard to generalizability of its results. The study uses a single-case design, studying how Data Governance effectiveness evolved at Johnson & Johnson. The conceptual framework also functions as the logic which links the data to the propositions (4th guiding point), and it provides the lens for analyzing and interpreting the findings (5th guiding point).

Johnson & Johnson represents a unique case according to the criteria proposed by Yin ([59], pp. 40–41) insofar as the company underwent almost an entire Data Governance lifecycle in a relatively short period of time. Furthermore, Johnson & Johnson has been frequently invited to present “best practice” in Data Governance at practitioners’ seminars and conferences. The case of Johnson & Johnson can therefore be considered as suitable for laying the foundations for further research.

Data was collected from different sources. The main data source, however, was four two-hour to eight-hour interviews with the director and the manager of Enterprise MDM at Johnson & Johnson. The interviews took place between September 26, 2011, and May 16, 2012. Transcripts of the interviews were created on the basis of the field notes taken by the researchers involved. The data was analyzed and interpreted according to the Business Engineering Case Study (BECS) method proposed by Senger and Österle [62]. The method was designed for research case studies in the field of Business Engineering and, thus, addresses the specific needs of data analyzing in the present study. The final case study report was sent to Johnson & Johnson for approval.

## 5 Data Governance at Johnson & Johnson

### 5.1 Company Overview

Johnson & Johnson is a Fortune 50 company and owns more than 250 subsidiaries located in 57 countries around the world. Johnson & Johnson is organized in three business segments, namely, “Pharmaceuticals,” “Medical Devices and Diagnostics,” “and Consumer Products.” With a staff of 114,000, Johnson & Johnson generated revenue of 65 billion USD in 2011. While business operations of “Pharmaceuticals” and “Medical Devices and Diagnostics” are directed by global organization, “Consumer Products” is divided into four geographical regions, namely, North America, South America, Asia-Pacific, and Europe<sup>3</sup>.

### 5.2 Initial Situation

#### 5.2.1 Strategic Perspective

Two of the goals which determine Johnson & Johnson’s business strategy are keeping a balanced product portfolio and growing through acquisitions. An example of a large takeover happening in the last years has been the acquisition of Pfizer’s consumer health care business in 2006. The deal was worth 16.6 billion USD.

Johnson & Johnson is focusing on its core competencies and has outsourced two thirds of its overall production.

#### 5.2.2 Business Process Perspective

Partly because of these extensive acquisition activities, business processes at Johnson & Johnson were not harmonized to a large extent in 2008 but differed widely across the numerous units and subsidiaries. For example, there were no company-wide guidelines for the pricing process.

Also, no shared understanding existed throughout the company of the definition of key business objects. “Product samples,” for example, were understood as products given away for free to customers by one business unit; the same term was used for promotional products which were handed to the sales personnel.

Not only did business processes differ across units, but there was also no company-wide data management department. Instead, across the various units, five major data management groups existed and worked independently of each other with no common purpose and goal.

---

<sup>3</sup>For more information on Johnson & Johnson’s Consumer Products division, see <http://www.jnj.com/connect/about-jnj/company-structure/consumer-healthcare>.

### 5.2.3 Information Systems Perspective

In 2005 a large project was started to implement SAP ERP across the company. The main objective was to introduce one standard application system for planning, production, sales, and distribution activities in Johnson & Johnson's "Consumer Products" business segment. The project included also software tools for managing master data creation.

Data management processes, though, such as the creation and maintenance of material master data were not fully defined when the system went "live" and were still organized locally with much variation. As a consequence, the investment in the project did not yield the expected return due to a weak link between the overall system and process design. Information systems for data management had not been deployed in a coordinated fashion.

### 5.2.4 Pain Points

Johnson & Johnson was suffering from a number of pain points. First, business process performance was poor in many areas. Customers were often sent incorrect invoices, trucks were waiting at the shipping docks for materials to be activated in the system, production was delayed at manufacturing plants, and purchase orders were not placed on time. Furthermore, the new product introduction process suffered from missing transparency about the status of new products, and no clear accountability for the management of the overall process was assigned.

Second, data quality was poor and caused problems when sending article information to data pools, such as the Global Data Synchronisation Network (GDSN)<sup>4</sup>. On top of that, Johnson & Johnson received critical feedback on the quality of logistics data, such as weight and dimensions of products, from one of its most important customers. In fact, the company was told when it came to logistics data quality it was worst among the strategic suppliers of that top customer.

Third, data management was not efficient. Approximately 80 % of the time was spent on investigating data quality defects and troubleshooting data quality issues.

## 5.3 Establishing Data Governance

### 5.3.1 Analysis Phase

Driven by the data quality-related pain points, Johnson & Johnson decided to thoroughly analyze its situation. In particular, the customer feedback on the poor

---

<sup>4</sup>See <http://www.gs1.org/gdsn> for details.

data quality regarding product weight and dimensions was taken up, and in 2008, a project was started together with the consulting branch of GS1 to “prove” the customer complaints.

GS1 introduced its CubiScan®<sup>5</sup> equipment to physically scan the products. The test took 1 month and every active product was scanned and analyzed. The result was eye-opening to the Johnson & Johnson management, as it turned out that for less than 30 % of products’ dimensions and weights, data was within the allowed 5 % error margin.

### 5.3.2 Founding Phase

In the second quarter of 2008, the executive management of Johnson & Johnson made the decision to set up a company-wide Enterprise Master Data (EMD) department, to ensure and manage data quality and prevent the company from experiencing bad business process performance due to poor data.

The designated EMD head was tasked with preparing the new organization within a time frame of 8 months. He was responsible for getting the commitment of the different business units to drop their local data management activities and integrate them into the new central EMD unit. The initiative was backed by executive management, so that the discussions could be done on the “how do we do it,” not on the “should we do it.” In this period of time, the past understanding of “my data”—i.e., data being owned by individual business units—was changed to “our data,” i.e., data being understood as an asset which belongs to the company. Eventually, the commitment for the joint activity was achieved on vice-president level within all units.

Key element of the founding phase was a so-called “kaizen” event, in which subject matter experts from all business units sat together in an offline location for 1 week to create a mutual understanding about the definition of key business objects and their use in different business processes. All major business functions, such as finance, engineering, or procurement, were represented there. The agenda stipulated that each day of the event, one specific business process, such as procurement or manufacturing, was to be discussed.

After a common understanding of key business objects had been achieved, roles and responsibilities were defined for the management and use of master data. Ownership was assigned for each and every field, by material type. The company ended up with a set of 420 master data attributes. Once ownership was assigned, the EMD team asked the owners to provide the rules of population. The team started off with the rules imposed by the systems in use (e.g., SAP). Then, business rules were developed. A simple example of a business rule is that every finished good needs a gross weight.

---

<sup>5</sup>See <http://www.cubiscan.com/> for details.

The activities during the founding phase were supported by external views on the topic. EMD managers from other consumer goods companies were invited to Johnson & Johnson to present their approaches and discuss achievements with the upper management.

### **5.3.3 Development Phase**

In May 2009 the new central EMD organization became operational, taking over responsibility for monitoring and ensuring that data of new products was ready on time and in the quality required. In the beginning, responsibilities for data objects were assigned per data class and business unit. However, over time people increasingly took over cross-unit responsibility, so that as of early 2010, there was fully regional responsibility for every data class in place.

Also the physical location of people underwent change. In the beginning, EMD members were located into common areas within each of the locations. Today the entire EMD team consisting of 27 people—16 are internal staff and 11 are external—is located at Johnson & Johnson’s headquarters.

The development phase was supported by yearly Master Data Summits and a steering committee on director level. The steering committee regularly evaluates the performance of the overall master data processes in terms of compliance to standards, quality targets, and on-time data delivery across the twelve business departments which are involved in the creation of master data.

Based on the new EMD unit, Johnson & Johnson also established company-wide data management processes. Core to these processes are a workflow-supported product data creation process and data quality monitoring.

### **5.3.4 Maturity Phase**

Starting in mid-2011, Johnson & Johnson has entered a mature level of its data management activities. Business processes for data management meanwhile have been widely adopted and accepted within the company, and central data management information systems have been continuously improved. As Data Governance has become increasingly effective, Johnson & Johnson is now able to provide product data on time and in better quality when new products are launched.

## **5.4 Current Situation**

### **5.4.1 Strategic Perspective**

The overall strategic goals of the company remained unchanged, because the focus is still growing through acquisitions and keeping a balanced product portfolio.

In contrast to the initial situation, though, executive management is now aware of the importance of master data quality for achieving strategic business goals. Thus, executive management took an active role in establishing and enforcing Data Governance—by deciding to establish an EMD department, for example.

### 5.4.2 Business Process Perspective

Today, business processes at Johnson & Johnson are using master data, such as product data, in a consistent way. The central EMD department has formed an unambiguous understanding about key data and has required business processes to adapt to data management needs in order to prevent the company from suffering poor data quality.

The EMD department is using business processes to ensure data quality before and during the use of the data, adopting lifecycle management concepts for physical goods to the purposes of data management.

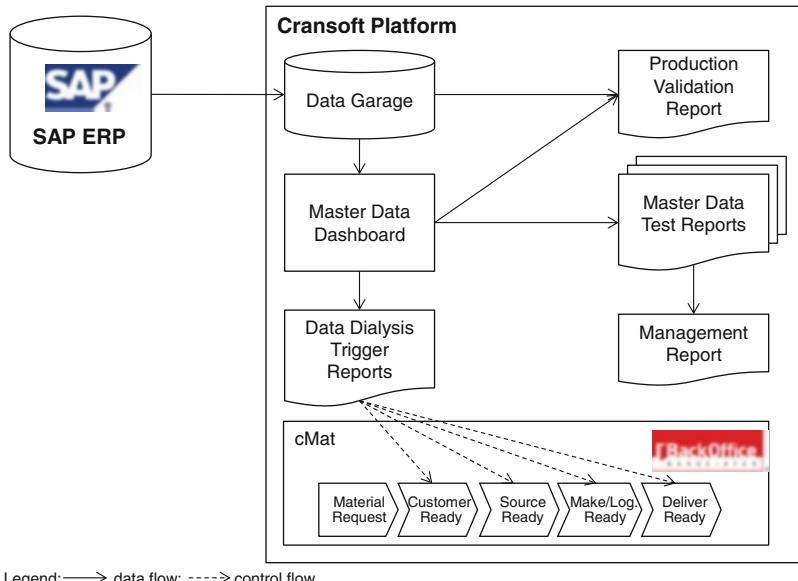
A prominent example of a business process ensuring data quality before data is being used is the process of introducing new products. While in the past there was no transparency of the product status, product data is now managed by a stage-gate process. Six months before a product is shipped to retailers, preliminary product data, such as packaging information, for example, is delivered to the retailers and to GS1. Three months before product shipment, final dimension and weight data is provided and the data is frozen in the SAP ERP system. Furthermore, a “packaging lab” has been installed in the headquarters and every new product is scanned before first shipment using the CubiScan equipment. As a result, verified product data can be ensured at the date of first shipment.

An example of a business process supporting data quality assurance while data is being used is the process of continuous analyzing, monitoring, and reporting of data quality of active goods.

Today the EMD department creates 3,000 master data records for finished goods and 11,000 master data records for raw materials—including raw material in the narrower sense and also work in progress, experimental material, and spare parts—per year. The current data quality level is 99.991 % (measured as the ratio of the number of error-free records to the total number of records in the SAP ERP system).

### 5.4.3 Information Systems Perspective

The newly established EMD team is using a number of information systems to achieve its goals. Figure 3 shows the information system landscape used to create material master data in the right quality and on time. The authoritative data source is the SAP ERP system, which is used company-wide. In addition to that, the EMD team has developed and is maintaining a set of systems to ensure data quality both when data records are created and when they are maintained. All systems are based on Back Office Associates’ Cransoft Platform.



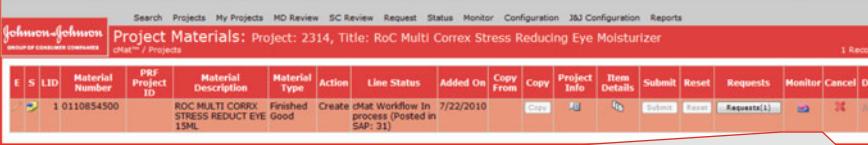
Legend: —→ data flow; - - -> control flow.

**Fig. 3** Information system landscape

Key systems are:

- Data Garage: Holds a daily copy of productive SAP data.
- Master Data Dashboard: Analysis and reporting tool for master data quality, which checks whether master data records violate the predefined set of rules and which reports those records that do not meet quality requirements.
- Trigger Reports: Information from the Master Data Dashboard together with the cMat workflow status controls the material master data creation flow. Only error-free materials are promoted to the next stage.
- cMat: Workflow management system supporting the creation of material master data records.
- Production Validation Report: Monitors critical field changes and errors for all existing commercialized materials which require immediate action.
- Master Data Test Reports: 350 different reports and queries can be produced by the Material Master Dashboard.
- Management Report: Management reports are used to highlight on-time and in-quality per department or other logical entity.

The information system landscape supports data quality assurance both before and while data is being used (see above). The cMat workflow management system is the central application system for ensuring that high-quality master data for finished goods and raw materials is provided on time. After the request of a new material it supports four different quality stages, namely, “customer ready,” “source ready,” “make ready,” and “delivery ready.” An illustrative example of a workflow status report displayed in the Material Master Dashboard is shown in Figure 4.



The screenshot shows a software interface for managing project materials. At the top, there's a navigation bar with links like Search, Projects, My Projects, MD Review, SC Review, Request, Status, Monitor, Configuration, J&J Configuration, and Reports. Below the navigation is a header bar with the company logo 'Johnson & Johnson GROUP OF CONSUMER COMPANIES' and the title 'Project Materials: Project: 2314, Title: RoC Multi Correx Stress Reducing Eye Moisturizer'. A sub-header indicates '1 Record'. The main area contains a table with columns: E, S, LID, Material Number, PRF Project ID, Material Description, Material Type, Action, Line Status, Added On, Copy From, Copy, Project Info, Item Details, Submit, Reset, Requests, Monitor, and Cancel. One row is shown with values corresponding to the project title.

E	S	LID	Material Number	PRF Project ID	Material Description	Material Type	Action	Line Status	Added On	Copy From	Copy	Project Info	Item Details	Submit	Reset	Requests	Monitor	Cancel
		1	0110854500		ROC MULTI CORRX STRESS REDUCT EYE 15ML	Good	Created	Workflow In process (Posted in SAP: 31)	7/22/2010							Requests(1)		

Milestone Phase	Milestone Object	DC Finished	Step Name	Source	Finished By	Finished On	Duration	Due On
Proj Initiation	PIO	✓	INITIATOR: Approved Project Request submitted	PI	E.Bender	7/26/2011	4 days	
Customer Ready	DP1	✓	PLANNING/REVIEW: Supply Chain Review complete	cMat	efarley	7/27/2011	1 day	7/29/2010
Customer Ready	MD1	✓	ENTERPRISE MASTER DATA: Workflow w/ Org Levels, AUM, Add'l Sale	cMat	MLawse	7/28/2011	1 day	7/31/2010
Customer Ready	DP1	✓	DEMAND PLANNING: DP Parent Code, Key Dates, Add'l Fcst Info	cMat				6/12/2011
Customer Ready	PK1	✓	PACKAGING: Draft Packaging Dimensions	cMat				6/12/2011
Customer Ready	SP1	✓	SUPPLY PLANNING: MRP Views, General Plant Data, Sales General Plan	cMat	EBender	10/4/2011	68 days	6/12/2011
Customer Ready	PU1	✓	SOURCING: PGroup, Matl Grp, Source List Ind, Order UOM	cMat	lignot4	7/28/2011	0 days	6/12/2011
Customer Ready	FI1	✓	FINANCE: Profit Center, Valuation Class, Costing and Accounting	cMat	klavasi1	7/29/2011	1 day	6/12/2011
Customer Ready	MD1		ENTERPRISE MASTER DATA: Post to status 31 CUSTOMER READY	cMat				6/13/2011
Source Ready	SP2	✓	SUPPLY PLANNING: Bill of Material (ECC)	SAP				8/15/2011
Source Ready	SP2		SUPPLY PLANNING: Production Version (ECC)	SAP				8/15/2011
Source Ready	PU2	✓	SOURCING: Purchasing Info Record (ECC)	SAP				8/15/2011
Source Ready	PU2		SOURCING: Outline Agreement (ECC)   Contract or Scheduling Agreements	SAP				8/15/2011
Source Ready	PU2		SOURCING: Source List (ECC)	SAP				8/15/2011
Source Ready	PU2		SOURCING: Update status 32 SOURCE READY	cMat				
Make Ready	GT3		GTO: Freight Classes, Commodity Info, HTS, Country of Origin posted	cMat				
Make Ready	QA3		QUALITY: GSS Spec Info posted	cMat				
Make Ready	PK3		PACKAGING: Final Packaging Dimensions entered	cMat				
Make Ready	EN3		ENVIRONMENTAL Haz Info, MSDS#, Disposal Req's, Recyclable Info	cMat				
Make Ready	FI3		FINANCE: Final Cost Rollup (ECC)	SAP				9/19/2011
Make Ready	QA3		QUALITY: Inspection Plans / Master Inspection Characteristics (ECC)	SAP				9/19/2011
Make Ready	QA3		QUALITY: Quality Info Record (ECC)	SAP				9/19/2011
Make Ready	MD3		ENTERPRISE MASTER DATA: Review AUM structure and post	cMat				
Make Ready	MD3		ENTERPRISE MASTER DATA: Post to 33 MAKE/LOGISTICS READY	SAP				9/19/2011
Deliver Ready	MD4		ENTERPRISE MASTER DATA: Sales BOM (ECC)	SAP				10/17/2011
Deliver Ready	FI4		MARKETING FINANCE: Material Pricing (ECC)	SAP				10/17/2011
Deliver Ready	MD4		ENTERPRISE MASTER DATA: Post to XDC31 DELIVERY READY	SAP				10/17/2011
Deliver Ready	MD4		ENTERPRISE MASTER DATA: Material Determination (ECC)	SAP				10/17/2011

Fig. 4 Material master creation workflow status report

## 5.5 Achievements and Success Factors

Within a time frame of 3 years Johnson & Johnson was able to achieve many Data Governance goals. The establishment of the EMD department paved the way toward the main functional goal, namely, to build up company-wide data management competencies. Based on the conceptual framework shown in Fig. 2, Table 2 shows the evolution of the six data management competencies. Five of the six competencies were newly built up on a company-wide level.

In addition to the functional goals, Johnson & Johnson made substantial progress with regard to achieving formal goals. Data Governance aims at increasing the quality of product data on the one hand and providing product data to the business processes at the right point of time on the other hand.

Much has been achieved with regard to data quality. Coming from a poor data quality level at which less than 30 % of the product data records were correct, the company has now achieved a “six sigma” level. As of July 1, 2012, 99.99966 % of all master data records comply with data quality rules (see Fig. 5).

The curve shows the evolution of the Data Quality Index at Johnson & Johnson. The index is computed as the ratio of the number of error-free material master records and the total number of material records. A record is considered defective

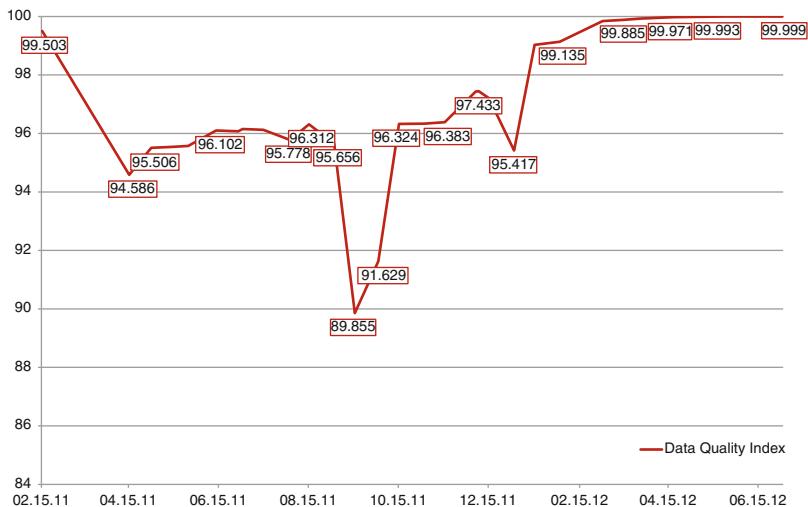
**Table 2** Evolution of data management competencies

Data management Competency	Q1/2008	Q1/2012
Data strategy management	Not existent on a company-wide level	EMD goals derived from business goals Commitment from executive management Continuous management reporting
Data quality management	Not existent at all	Data quality processes and tools in place and continuously improved both for quality assurance before and during data use
Data stewardship	On business unit level only No company-wide coordination	Clear responsibilities for data classes Central EMD team with stewardship responsibility
Data lifecycle management	No company-wide approach	Workflow-supported creation and maintenance of data records Master data deactivation process underdevelopment
Data architecture management	No common definition of key business objects  Central data source through SAP ERP	Shared, unambiguous understanding of key business objects (e.g., product samples) SAP ERP authoritative data source
Database management	SAP ERP	SAP ERP

if it violates one of the validation rules related to material master. The first set of validation rules was identified in the Data Governance founding phase (see above). Today, Johnson & Johnson has approximately 400 validation rules which are extended or adjusting on a daily basis. The constant revision of rules is a response to the market dynamics of the consumer product business in North America.

The curve shows two significant drops. The first one on September 15, 2011, was a consequence of a migration of data from an additional company to the SAP platform. The second drop in early January 2012 was simply the result of the measuring point of time chosen. The measuring took place on January 1, 2012, which was a Sunday. At this time, the calendar year did not match the fiscal year—a mismatch which resolved itself the next Monday, January 2. Testing logic, though, was adjusted to prevent this error from happening in the future.

Johnson & Johnson has identified a number of success factors which had a positive impact on the establishment of effective Data Governance within its organization. First, it is today considered fundamentally important that the main impulse to start Data Governance activities came from outside the company. While internal drivers of data quality management did exist, awareness of the problem was not sufficient, and so these issues were not given top priority when it came to assigning resources to improve the situation. The decisive impulse had to come from the customer side, as “internal problems didn’t matter.”



**Fig. 5** Evolution of the Data Quality Index

Second, the severity of the issue caused immediate and significant action. The customer complaint was considered to be the “big wake-up call” that set things in motion. This incident helped overcome the problem that poor data quality was widely being perceived as unrelated, small issues occurring at different times in different places in the company, i.e., issues which did not require company-wide action. Instead, executive management became aware of the company-wide implications of data quality and made a top-down decision for Data Governance and to establish the EMD department.

Third, monitoring data quality on a daily basis and tracking the results in time allow Johnson & Johnson to visualize the impact of certain business event on data quality. As a result, the company can prepare for future business events, reducing the impact on data quality. Apart from that, measuring data quality and monitoring its improvement up to six sigma level demonstrate that effective Data Governance actually has a positive impact on data quality. This effect helps justifying the continuous effort of not only establishing Data Governance but driving its evolution over a significant amount of time.

## 6 Interpretation of Findings

### 6.1 Data Governance as a Dynamic Capability

The case of Johnson & Johnson supports the notion of Data Governance being a dynamic capability. In this case, Data Governance was a reaction to a significant

change in the company's environment. A customer complaint triggered activities to change settled procedures, initiating a transformation process in the company. The resulting rearrangement of resources took place on all three Business Engineering levels, involving creation, integration, and reconfiguration processes.

On a strategic level, data quality has become a new strategic goal which is continuously reported to executive management. Furthermore, the way data is perceived in the company has changed. While in the past data was seen as belonging to individual units or even employees, Johnson & Johnson today understands data as a company asset and has moved from "my data" to "our data" approach.

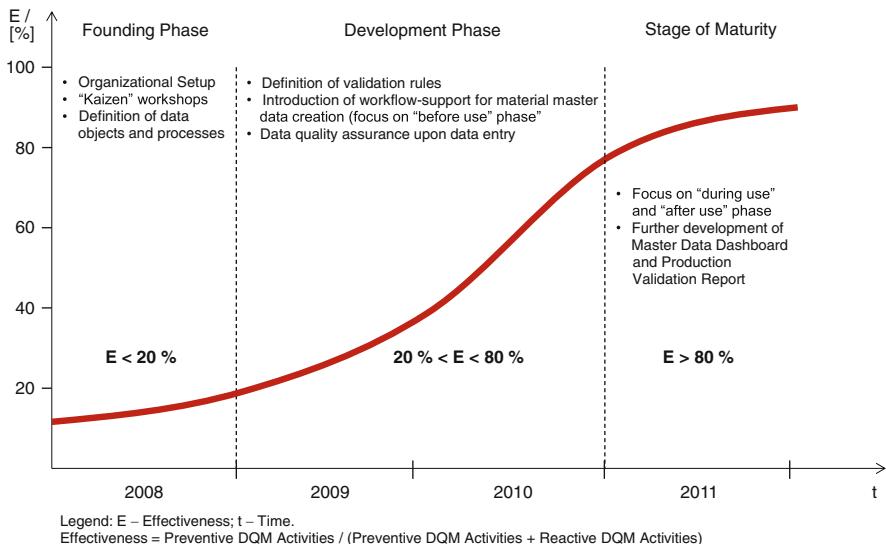
On an organizational level, a new department, namely, the EMD unit, was created. Headed by an employee on director level, the EMD unit reports to supply chain management and at present employs twenty internal and ten external staff. Team members were integrated into one department and also "centralized" in one physical location at the company's headquarters. Furthermore, business processes were reconfigured to ensure high data quality from creation to use of master data.

On the information systems level, a landscape of systems was introduced to support EMD activities. While in the past SAP ERP was the only application system for managing master data, the company today employs a set of configurable software tools which support data quality assurance both before and while data is being used.

Apart from that, the case provides new insight into the evolution of Data Governance effectiveness. Data Governance effectiveness can be interpreted as the ratio of the number of preventive DQM activities to the total number of DQM activities. A value close to 1, for example, indicates high Data Governance effectiveness, with almost all DQM activities being preventive. The theory of the capability lifecycle predicts an evolution of dynamic capabilities starting with a founding phase, moving on to a development phase, and resulting in a phase of maturity, meaning a high degree of effectiveness [49]. Current research concedes that further research is needed to better understand the nature of the maturity evolution curve of capabilities.

Data Governance effectiveness in the case of Johnson & Johnson has evolved according to an S-shaped curve (see Fig. 6). In the beginning in 2008, more than 80 % of all DQM activities were reactive in nature. Even when the EMD department became operational in early 2009, the EMD staff had to deal a lot with correcting data errors. Typical effects of these data errors in business processes were delays in goods entry, because a data record on raw material was not available, and purchase orders which could not be issued because finished goods data was incomplete.

With the introduction of cMat workflow support for master data creation in the development phase, Johnson & Johnson was able to ensure that no more defective data entered the system. The workflow continuously increased the proportion of data in the system which was available on time and in the required quality. As a result, the EMD department was able to free up more and more capacity to work on preventive DQM activities. No additional people needed to be hired, but existing personnel could increasingly be assigned to developing further the workflow support, defining validation rules for material master data, and establishing reporting tools, such as the Master Data Dashboard.



**Fig. 6** Data Governance effectiveness

The cMat workflow is a cornerstone of the so-called “1st time right” philosophy, which aims at ensuring data quality upon data entry. In addition to that, the later introduced Production Validation Report (PVR) aims at continuously improving the quality of active material master data records already existent in the system. The combination of both approaches enabled Johnson & Johnson to reach a stage of maturity in which as of the end of 2011 less than 10 % of all DQM activities have been reactive. Thus, Data Governance effectiveness is at 90 %, approximately.

## 6.2 Managing Data Governance Effectiveness

Research in the areas of both IT Governance [63, 64] and Data Governance [1, 28] has shown that Data Governance arrangements are contingent on a set of company internal and external factors. Typical external factors are, for example, the specific characteristics and processes of the industry segment a company is operating in or certain market dynamics a company is exposed to. Examples of internal contingency factors are the company’s governance approach, its business and IT strategy, and the level of business process harmonization it has achieved.

These findings suggest that not only Data Governance arrangements but also its evolution over time is contingent on such factors. The case of Johnson & Johnson shows an S-shaped curve of Data Governance effectiveness, i.e., Data Governance effectiveness could steadily be improved. However, the curve might look different in companies with different contingency factors. Johnson & Johnson

had an “eye-opening” event in 2008 when confronted with complaints about poor logistics data quality from one of its most important customers—which paved the way for Data Governance being supported by executive management. Other cases show a different evolution. British Telecom, for example, started with a small data quality task force and developed Data Governance capabilities further over a period of almost 10 years (from 1997 until 2006) [28].

The assumption of Data Governance evolution being contingent on a number of factors is supported by Teo and King [65], who investigated the combined application of contingency and evolutionary theory in the field of integrating business and information systems planning.

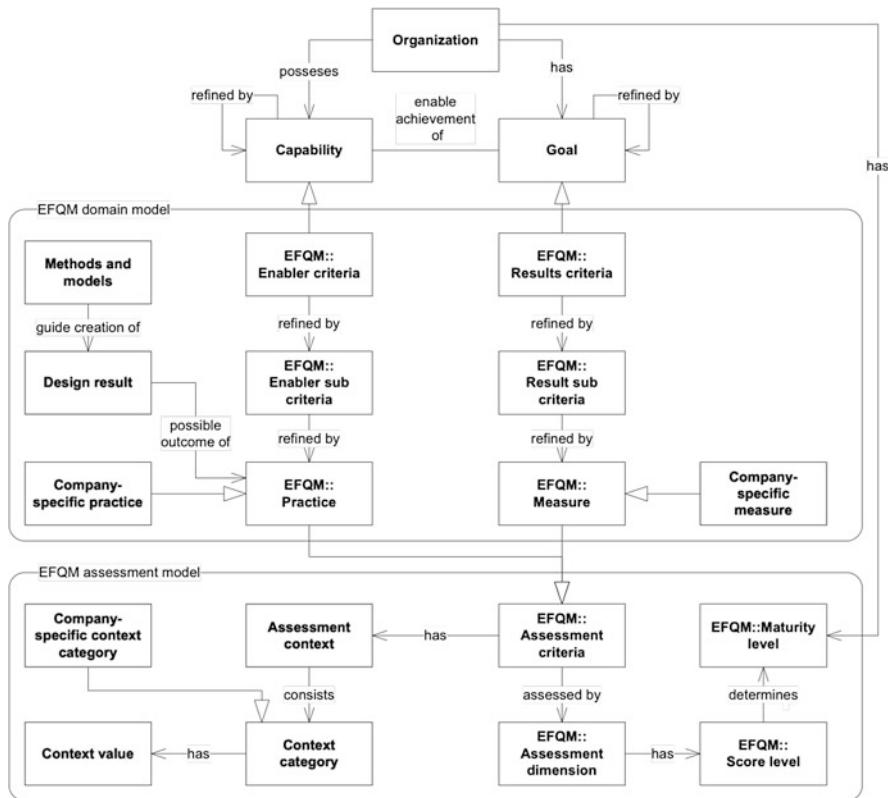
### 6.3 *Maturity Model for Data Governance Effectiveness*

As Data Governance effectiveness evolves over time, the question arises as to how this evolution can be managed and monitored. An appropriate means to manage the establishment of organizational capabilities are maturity models. Maturity models represent a special class of models, dealing exclusively with organizational and/or information systems related to change and development processes [66–68]. Maturity models consist of an organized set of constructs serving to describe certain aspects of maturity of a capability to be designed [69].

Typically, a maturity model consists of a domain model and an assessment model. The domain model comprises criteria by which the design domain can be partitioned into discrete units to be assessed. The assessment model provides one or multiple assessment dimensions each defining an assessment scale. What is basically assessed is to what extent certain criteria comply with the scale for each assessment dimension. In order to be able to structure the assessment process, some maturity models also provide appraisal methods (e.g., Standard CMMI Appraisal Method for Process Improvement, SCAMPI) [70].

A number of maturity proposals can be found in the fields of DQM and Data Governance. Lee et al. [71] have proposed a methodology called AIM quality (AIMQ), which can be used as a basis for information quality assessment and benchmarking. This methodology uses 65 criteria to evaluate results to be achieved by DQM. DataFlux [72] has come up with a maturity model comprising four criteria (people, policies, technology, and risk and reward) by which companies can assess the progress of DQM establishment in their organization. Gartner [73] aims at the same objective with their maturity model, using quite vague definitions of individual levels of maturity instead of discrete criteria. Furthermore, Ryu et al. [74] and Baskarada et al. [75] have developed maturity models on the basis of the Capability Maturity Model Integration (CMMI) approach [76].

None of these maturity models covers all aspects of Data Governance in their entirety though. Guidelines for designing actions for improvement are offered by two approaches only. Also, all maturity models examined are characterized by a rigid, predefined path of development [57, 58]. This, however, stands in contrast



**Fig. 7** Meta-model of the EFQM Excellence Model for corporate data quality management

with the view of DAMA [19], stating—in line with the assumption of Data Governance being contingent on a number of factors—that “[...] how each enterprise implements [data quality management] varies widely. Each organization must determine an implementation approach consistent with its size, goals, resources, and complexity. However, the essential principles of [data quality management] remain the same across the spectrum of enterprises [...].” Taking this into account, a maturity model must provide a dynamic path of development, which each organization can adapt to its individual needs and requirements.

This design principle was taken up when the EFQM Excellence Model for Corporate Data Quality Management was developed [77]. The model was jointly developed by EFQM and the Competence Center Corporate Data Quality (CC CDQ). The CC CDQ is a consortium research project at the University of St. Gallen, in which more than twenty large enterprises from various industries have participated since 2006.

Figure 7 shows the meta-model illustrating the conceptual elements of the maturity model. The modeling language used is *Meta-Object Facility (MOF)* [78],

which is a meta-language used, for example, for specifying UML, and which supports transformation of conceptual models into code that can be implemented. Model elements adopted from the EFQM Excellence Model are indicated with the EFQM namespace prefix. The maturity model is built upon the logic that each organization defines goals for Data Governance and requires certain capabilities to achieve them. Therefore, the model uses so-called *Enablers* and *Results* criteria to assess an organization's Data Governance maturity. At its core, the maturity model defines 30 *Practices* and 56 *Measures* for DQM that are used as assessment elements during an appraisal. To give examples, "Running an adequate enterprise DQM training program to develop people's knowledge and competencies regarding their current and future needs to manage enterprise data" is a *Practice*, whereas "Success rate of enterprise data quality related training and development of individuals" is a *Measure*. For a complete list, see EFQM [77]. *Practices* are used to assess if and how well certain DQM capabilities are already established in an organization, whereas *Measures* allow to assess if and how well the *Practices* support the achievement of goals.

The maturity model allows companies to continuously assess and monitor their current Data Governance maturity level. Thus, the model can be used to plan and control the evolution of Data Governance over time and to identify need for actions—in case effectiveness is decreasing or stagnating.

## 7 Conclusions

The chapter presents a single-case study at Johnson & Johnson to explore the evolution of Data Governance in large enterprises. The case study suggests that Data Governance can be seen as a dynamic capability, i.e., as an organization's ability to rearrange its resources as a response to a market change in order to keep its competitive position. Johnson & Johnson created, integrated, and reconfigured strategic, organizational, and information systems resources in order to ensure the quality of its material data and, by doing so, improve its business processes.

Interpretation of the findings of the case suggests that Data Governance evolution—as well as Data Governance arrangements—is contingent on both external and internal factors. Managing Data Governance evolution, thus, requires awareness of the contingency factors and their influence in company-specific environments. A tool to support this continuous monitoring and management process is the EFQM Excellence Model for Corporate Data Quality Management.

The scientific contribution mainly stems from the fact that it represents the first effort of investigating in detail the evolution of Data Governance effectiveness. It thereby lays the ground for a more detailed understanding of the nature of the effectiveness of Data Governance and its evolution. Furthermore, it advances the understanding of capability lifecycles as stipulated by [49], as it provides detailed insight about the phase-wise evolution of a dynamic capability.

Practitioners can benefit from the study, too. The case gives detailed indication for the establishment of Data Governance in large enterprises. The findings are applicable to a variety of different organizations.

Future research should take up on the results and conduct further qualitative and quantitative studies to deepen the knowledge about the nature of the Data Governance evolution curve and the impact of the various contingency factors.

## References

1. Weber K, Otto B, Österle H (2009) One size does not fit all—a contingency approach to data governance. *ACM J Data Inf Qual* 1(1):Article 4
2. Khatri V, Brown CV (2010) Designing data governance. *Commun ACM* 53(1):148–152
3. IBM (2011) IBM Forms Data Governance Council 2005, February 9, 2011. <http://www-03.ibm.com/press/us/en/pressrelease/7766.wss#release>
4. AFNOR (2009) General principles for an environmental communication on mass market products. 2009, AFNOR Groupe: La Plaine Saint-Denis, France
5. Deloitte (2009) Telecommunications Predictions: TMT Trends 2009. Deloitte Touche Tohmatsu, London
6. Otto B (2011) A morphology of the organisation of data governance. In: 19th European Conference on Information Systems, Helsinki, Finland
7. Pierce E, Dismute WS, Yonke CL (2008) The state of information and data governance—understanding how organizations govern their information and data assets. IAIDQ and UALR-IQ, Little Rock, AR
8. Badenoch D et al (1994) The value of information. In: Feeney M, Grieves M (eds) *The value and impact of information*. Bowker-Saur, London, pp 9–75
9. Boisot M, Canals A (2004) Data, information and knowledge: have we got it right? *J Evol Econ* 14(1):43–67
10. Oppenheim C, Stenson J, Wilson RMS (2003) Studies on information as an asset I: definitions. *J Inf Sci* 29(3):159–166
11. Van den Hoven J (1999) Information resource management: Stewards of data. *Inf Syst Manag* 16(1):88–91
12. Holtham C (1995) Resolving the imbalance between information and technology. In: Best D (ed) *The fourth resource: information and its management*. Aslib/Gower, Aldershot, UK, pp 41–58
13. Wang RY (1998) A product perspective on total data quality management. *Commun ACM* 41(2):58–65
14. Olson J (2003) Data quality—the accuracy dimension. Morgan Kaufmann, San Francisco
15. Redman TC (2001) Data quality. The field guide. Digital Press, Boston
16. Wang RY, Strong DM (1996) Beyond accuracy: what data quality means to data consumers. *J Manag Inf Syst* 12(4):5–34
17. Horne NW (1995) Information as an asset—the board agenda. *Comput Audit Update* 1995(9):5–11
18. Even A, Shankaranarayanan G (2007) Utility-driven assessment of data quality. *ACM SIGMOD Database* 38(2):75–93
19. DAMA (2009) *The DAMA guide to the data management body of knowledge*. Technics Publications, Bradley Beach
20. Helfert M (2002) *Proaktives Datenqualitätsmanagement in Data-Warehouse-Systemen: Qualitätsplanung und Qualitätslenkung* (Germ.: Proactive data quality management in data warehouse systems: Quality planning and controlling). Logos, Berlin

21. Batini C et al (2009) Methodologies for data quality assessment and improvement. *ACM Comput Surv* 41(3):1–52
22. Wang RY et al (1998) Manage your information as a product. *Sloan Manag Rev* 39(4):95–105
23. ISO/IEC (2008) ISO/IEC 38500: corporate governance of information technology. ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission), Geneva, Switzerland
24. Sadiq S, Yeganeh K, Indulská M (2011) Cross-disciplinary collaborations in data quality research. In: 19th European Conference on Information Systems, Helsinki, Finland
25. Kooper M, Maes R, Lindgreen ER (2009) Information governance: in search of the forgotten grail. In: PrimaVera Working Paper Series. University of Amsterdam, Amsterdam, The Netherlands
26. Cragg P, Caldeira M, Ward J (2011) Organizational information systems competences in small and medium-sized enterprises. *Inf Manag* 48(8):353–363
27. Lucas A (2010) Corporate data quality management in context. In: 15th International Conference on Information Quality, Little Rock, AR
28. Otto B (2011) Organizing data governance: findings from the telecommunications industry and consequences for large service providers. *Commun AIS* 29(1):45–66
29. Wende K, Otto B (2007) A contingency approach to data governance. In: 12th International Conference on Information Quality, Cambridge, MA
30. Cheong LK, Chang V (2007) The need for data governance: a case study. In: Toleman M, Cater-Steel A, Roberts D (eds) 18th Australasian Conference on Information Systems. The University of Southern Queensland, Toowoomba, Australia, pp 999–1008
31. Vaygan JA et al (2007) The internal information transformation of IBM. *IBM Syst J* 46(4):669–683
32. Beynon-Davies P (2005) Personal identification in the information age: the case of the national identity card in the UK. In: 13th European Conference on Information Systems, Regensburg, Germany
33. Delbaere M, Ferreira R (2007) Addressing the data aspects of compliance with industry models. *IBM Syst J* 46(2):319–334
34. Ramakrishnan T, Jones MC, Sidorova A (2012) Factors influencing business intelligence (BI) data collection strategies: an empirical investigation. *Decis Support Syst* 52(2):486–496
35. Smith HA, McKeen JD (2008) Developments in practice XXX: master data management: salvation or snake oil? *Commun AIS* 23:63–72
36. Williams CB, Fedorowicz J, Tomasino AP (2010) Governmental factors associated with statewide interagency collaboration initiatives. In: 11th Annual International Conference on Digital Government Research, Puebla, Mexico, pp 14–22
37. Panian Z (2010) Some practical experiences in data governance. *World Acad Sci Eng Technol Manag* (62):939–946
38. Ardagna CA et al (2009) An XACML-based privacy-centered access control system. In: 1st ACM Workshop on Information Security Governance, Chicago, IL, pp 49–57
39. Dan A, Johnson R, Arsanjan A (2007) Information as a service: modeling and realization. In: International Workshop on Systems Development in SOA Environments
40. Gates C, Bishop M (2010) The security and privacy implications of using social networks to deliver healthcare. In: 3rd International Conference on Pervasive Technologies Related to Assistive Environments, Samos, Greece
41. Kerschbaum F, Schaad A (2008) Privacy-preserving social network analysis for criminal investigations. In: 7th ACM Workshop on Privacy in the Electronic Society, Alexandria, VA, pp 9–13
42. Pearson S (2009) Taking account of privacy when designing cloud computing services. In: ICSE Workshop on Software Engineering Challenges of Cloud Computing, Vancouver, Canada, pp 44–52
43. Stell A, Sinnott R, Ajayi O (2008) Supporting UK-wide e-clinical trials and studies. In: 15th ACM Mardi Gras Conference, Baton Rouge, LA

44. Rifaie M, Alhajj R, Ridley M (2009) Data governance strategy: a key issue in building enterprise data warehouse. In: 11th International Conference on Information Integration and Web-based Applications & Services, Kuala Lumpur, Malaysia, pp 587–591
45. Gillies A, Howard J (2005) An international comparison of information in adverse events. *Int J Health Care Qual Assur* 18(5):343–352
46. Rosenbaum S (2010) Governance data and stewardship, designing data stewardship entities and advancing data access. *Health Serv Res* 45(5):1442–1455
47. Wernerfelt B (1984) A resource based view of the firm. *Strateg Manag J* 5(2):171–180
48. Barney J (1991) Firm resources and sustained competitive advantage. *J Manag* 17(1):99–120
49. Helfat CE, Peteraf MA (2003) The dynamic resource-based view: capability lifecycles. *Strateg Manag J* 24(10):997–1010
50. Dierickx I, Cool K (1989) Asset stock accumulation and sustainability of competitive advantage. *Manag Sci* 35(12):1504–1511
51. Cepeda G, Vera D (2007) Dynamic capabilities and operational capabilities. *J Bus Res* 60(5):426–437
52. Eisenhardt KM, Martin JA (2000) Dynamic capabilities: what are they? *Strateg Manag J* 21(10–11):1105–1121
53. Wang CL, Ahmed PK (2007) Dynamic capabilities: a review and research agenda. *Int J Manag Rev* 9(1):31–51
54. Teece DJ, Pisano G, Shuen A (1997) Dynamic capabilities and strategic management. *Strateg Manag J* 18(7):509–533
55. Österle H (1996) Business engineering: transition to the networked enterprise. *Electron Mark* 6(2):14–16
56. Österle H, Blessing D (2003) Business engineering modell. In: Österle H, Winter R (eds) *Business Engineering*. Springer, Berlin, pp 65–85
57. Hüner K, Ofner M, Otto B (2009) Towards a maturity model for corporate data quality management. In: Shin D (ed) 2009 ACM Symposium on Applied Computing. Honolulu, HI, pp 231–238
58. Ofner M, Hüner K, Otto B (2009) Dealing with complexity: a method to adapt and implement a maturity model for corporate data quality management. In: 15th Americas Conference on Information Systems, San Francisco, CA
59. Yin RK (2002) *Case study research: design and methods*, 3rd edn. Sage Publications, Thousand Oaks, CA
60. Eisenhardt KM (1989) Building theories from case study research. *Acad Manag Rev* 14(4):532–550
61. Benbasat I, Goldstein DK, Mead M (1987) The case research strategy in studies of information systems. *MIS Q* 11(3):369–386
62. Senger E, Österle H (2004) PROMET Business Engineering Case Studies (BECS) Version 2.0. University of St. Gallen, Institute of Information Management, St. Gallen
63. Weill P, Ross J (2005) A matrixed approach to designing IT governance. *MIT Sloan Manag Rev* 46(2):25–34
64. Sambamurthy V, Zmud RW (1999) Arrangements for information technology governance: a theory of multiple contingencies. *MIS Q* 23(2):261–290
65. Teo TSH, King WR (1997) Integration between business planning and information systems planning: an evolutionary-contingency perspective. *J Manag Inf Syst* 14(1):185–214
66. Nolan RL (1973) Managing the computer resource: a stage hypothesis. *Commun ACM* 16(7):399–405
67. Gibson CF, Nolan RL (1974) Managing the four stages of EDP growth. *Harv Bus Rev* 52(1):76–88
68. Crosby PB (1980) *Quality is free: the art of making quality certain*. Mentor, New York
69. Fraser P, Moultrie J, Gregory M (2002) The use of maturity models/grids as a tool in assessing product development capability. In: IEEE International Engineering Management Conference, Cambridge, UK

70. SEI (2006) Standard CMMI Appraisal Method for Process Improvement (SCAMPI[SM]) A, Version 1.2: Method Definition Document. Carnegie Mellon University, Pittsburgh
71. Lee YW et al (2002) AIMQ: a methodology for information quality assessment. *Inf Manag* 40:133–146
72. DataFlux (2007) The Data Governance Maturity Model. DataFlux Corporation, Cary
73. Bitterer A (2007) Gartner's Data Quality Maturity Model. Gartner, Stamford
74. Ryu K-S, Park J-S, Park J-H (2006) A data quality management maturity model. *Electron Telecommun Res Inst J* 28(2):191–204
75. Baskarada S, Gao J, Koronios A (2006) Agile maturity model approach to assessing and enhancing the quality of asset information in engineering asset management information systems. In: Abramowicz W, Mayr HC (eds) 9th International Conference on Business Information Systems, Klagenfurt, Austria, pp 486–500
76. SEI (2006) CMMI for Development, Version 1.2. Carnegie Mellon University, Pittsburgh
77. EFQM (2012) EFQM Framework for Corporate Data Quality Management. EFQM, Brussels
78. OMG (2006) Meta Object Facility (MOF) Core Specification—Version 2.0, 1 November 2011. <http://www.omg.org/spec/MOF/2.4.1/PDF/>

## **Part II**

# **Architectural Aspects of Data Quality**

The technology landscapes for deployment of data quality solutions will be presented in this part. However, technology architectures have constantly evolved since the time of mainframes. These include centralized, client–server, peer-to-peer, service oriented architectures, and now more recently those based on multi-core, in-memory, and cloud computing, to name a few. Since each is a topic of study by itself, focus in this part is to highlight some key architectures and how they relate to the support of data quality management.

The first chapter in this part discusses data quality in data warehouses. In fact data warehouses and various decision support platforms were one of the earliest triggers for identification of data quality problems (see a brief history of the data quality discipline presented in the final chapter of the handbook). Lukasz Golab provides a survey of data quality problems in this chapter as well as some of the notable solutions relating to data freshness, completeness, and consistency.

The second chapter by Martin Hepp and Christian Fuerber highlights the role of semantic web technologies in data quality management. Some important uses and their limitations have been discussed, which include semantic definition of data, creation of shared understanding, and facilitation of content integration, as well as collaborative representation and use of quality-relevant knowledge.

Data (quality) profiling tools are an essential component of technology solutions and platforms for data quality management. Data profiling and providing means of measuring data quality is a vast area and has been addressed from both the data model (or meta-data) and the data population perspectives. Conceptual model quality is a significant area of study and there is a complementary body of knowledge in the domain. Hence, this chapter focuses on error detection in data populations using advanced statistical techniques. Metrics for assessing data quality as well as strategies for cleaning are discussed.

# Data Warehouse Quality: Summary and Outlook

Lukasz Golab

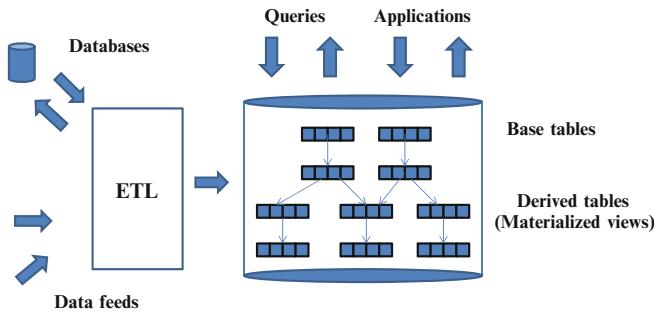
**Abstract** Data warehouses correlate data from various sources to enable reporting, data mining, and decision support. Some of the unique features of data warehouses (as compared to transactional databases) include data integration from multiple sources and emphasis on temporal, historical, and multidimensional data. In this chapter, we survey data warehouse quality problems and solutions, including data freshness (ensuring that materialized views are up to date as new data arrive over time), data completeness (capturing all the required history), data correctness (as defined by various types of integrity constraints, including those which govern how data may evolve over time), consistency, error detection and profiling, and distributed data quality.

## 1 Introduction

A data warehouse is a collection of data gathered from various structured and unstructured sources, such as transactional databases, Web sites, documents, spreadsheets, social media, and externally generated data feeds. Once data are loaded, data warehouses are used for read-only purposes, such as ad hoc querying, business reporting, data mining, historical what-if analysis, and decision support. A data warehouse system is a data management system with built-in features to facilitate complex queries over very large data sets, including indexes, materialized views, data compression techniques, data structures optimized for storing a long history and appending new data (as opposed to modifying existing data), and SQL extensions for expressing complex analyses beyond standard group-by aggregates [30]. Poor data warehouse quality is a serious and costly issue: if we cannot trust the data, we cannot trust the knowledge extracted from the data. In this chapter, we survey

---

L. Golab (✉)  
University of Waterloo, Waterloo, ON, Canada  
e-mail: [lgolab@uwaterloo.ca](mailto:lgolab@uwaterloo.ca)



**Fig. 1** An abstract architecture of a data warehouse system

common data warehouse quality problems and techniques that data warehouse systems and users can use to address these problems.

Figure 1 illustrates an abstract architecture of a typical data warehouse system. Data may arrive from multiple sources with various frequencies, ranging from daily extracts of operational databases to nearly real-time data feeds producing new data every few minutes. Data feeds are typically *pushed* to the warehouse by the data sources, while database extracts may be pushed or *pulled* by the warehouse system. The Extract–Transform–Load (ETL) process pulls (if necessary) and converts raw data into a format expected by the warehouse; this process may reside on a different machine, and it may use a separate staging database. Some transformations may be done inside the warehouse after loading the data, leading to the Extract–Load–Transform (ELT) model. Simple data quality checks such as record-level predicates (e.g., ensuring that Age and Salary values are nonnegative) may be done at the Transform step.

In traditional data warehouses, updates are typically applied during downtimes, e.g., every night [26]. In contrast, a *stream* data warehouse [26], also known as an active or real-time data warehouse, attempts to load new data feeds as they arrive so that applications that depend on the data can take immediate action. In practice, a large data warehouse may contain a mix of “streaming” and periodically refreshed tables. However, even stream warehouses perform batch updates for efficiency, in contrast to Data Stream Management Systems (DSMSs) [26] that perform record-at-a-time processing.

A data warehouse generally stores two kinds of tables. *Base tables* are sourced from the ETL process, which takes in the raw input data. *Derived tables* correspond to materialized views, which are precomputed results of SQL queries or external scripts or programs over one or more base or other derived tables.

Base tables may be divided into fact and dimension tables. Fact tables store various measure attributes (“facts”); for example, a retailer may keep track of the dollar amount of each customer transaction. Dimension tables store multiple, often hierarchical, attributes that describe the objects being measured (via foreign key relationships between fact and dimension tables), such as the location of each store

(city district, city, region, province, country, etc.) or the type of product sold (item name, category, manufacturer, etc.). Derived tables may summarize base data at different granularities, precompute join results (e.g., of fact and dimension tables), or maintain statistics or models over measure attributes, grouped by various subsets of dimension attributes. Fact tables may be very large and may change frequently (in an append-only manner) as new facts arrive. Dimension tables also change over time, but less frequently; for example, products may be reclassified and customers may change their addresses or the services they subscribe to.

In a traditional data warehouse system, the ETL process invokes periodic updates of the entire warehouse, starting with base tables and working through the materialized view hierarchy. Derived tables may be updated incrementally if possible or recomputed from scratch. In a stream warehouse, data arrive asynchronously, and the corresponding base tables should be updated upon arrival of a batch of new data. After a base table has been updated, any derived tables defined directly on it are now due for an update; when those derived tables have been updated, any derived tables defined directly on them may be updated, and so on. In order to keep up with the inputs, view updates in stream warehouses must be performed incrementally [21].

Usually, data in a warehouse are temporal (i.e., at least one of dimension attributes is a time attribute) and insert-only. For example, we may receive a daily data feed with point-of-sale purchase transactions from various stores, in which each purchase is associated with a timestamp and previously arrived purchase records do not change in the future. A common way to organize historical tables is to horizontally partition them according to a timestamp attribute, with each partition being indexed separately. This way, when new data arrive, new partitions are created as necessary to store the data; older partitions and any indexes defined on them are not affected by updates and do not have to be modified. If the warehouse is running out of space, old partitions may be deleted, aggregated, or otherwise compressed.

## 1.1 Sources of Data Warehouse Quality Problems

Since data warehouses collect data from transactional databases, they may inherit the same types of data quality problems. These include incorrect data (e.g., due to data entry errors), inconsistent data (e.g., using the same value to represent different things over time), hidden default and missing values (e.g., 99,999 zip codes), misplaced data, and violations of record-level, table-level, and cross-table integrity constraints (e.g., functional or conditional functional dependencies [13] may not be strictly enforced at the source database). Additional problems also arise due to the heterogeneous and temporal nature of a data warehouse, as outlined below.

The first issue, related to the ETL process, is that of *data integration*, i.e., extracting, correlating, and merging data from multiple, sometimes poorly documented, sources. This is a challenging topic with a lengthy literature. For the purposes of this chapter, we assume that data integration issues, such as finding the most accurate

and trustworthy sources for various types of data, duplicate removal, and schema and data formatting inconsistencies across sources, have already been addressed.

The second issue is that in addition to extracts from transactional databases, data warehouses may ingest data feeds generated from a wide variety of external sources: point-of-sale purchase transactions, financial tickers, trouble tickets, system logs, sensor measurements, social media, etc. Some of these data feeds may be machine generated and may contain new kinds of errors due to the inherent *imprecision* of the sources (e.g., sensors) or due to hardware and software problems at the sources or at the data collecting mechanism. Furthermore, *data completeness* problems may arise: data may be missing (not collected for various reasons or lost en route to the warehouse) or delayed, and it may not be clear if and when all the expected data have arrived.

The third issue, which affects stream warehouses, arises from the temporal nature of data feeds. Since new data continually arrive over time, the warehouse must be continually updated to ensure *data freshness*. Even if any data quality issues have been resolved, the utility of a warehouse suffers if data are not available to users and applications in timely fashion. On the other hand, updating a base table or a materialized view too soon may produce *inconsistent* and unrepeatable results. For example, suppose we have a fact table with individual sales that arrive continuously throughout the day and a materialized view that computes daily totals for each department or product category. If we update the view in the middle of the day, today's daily totals will be incomplete. Users who query the view at different times of the day may see different results.

Fourth, problems may arise due to the need to store a long history. Examples include data availability (we may have to delete old data to save space, but someone may need these data in the future) and historical completeness (historical analyses may not be possible if data are missing for a period of time). Moreover, we may need to define new integrity constraints for data evolution over time (e.g., a never-married person may become married, but not vice versa).

Finally, large data warehouses are usually distributed across multiple machines or even multiple data centers, making inconsistency detection and correction challenging.

## 1.2 Roadmap

In the remainder of this chapter, we take a closer look at the problems listed above. In Sect. 2, we start with a general architecture for characterizing and improving data warehouse quality. In Sect. 3, we discuss update strategies for maximizing data freshness. In Sect. 4, we address data currency, i.e., ensuring that data are not obsolete. In Sects. 5 and 6, we discuss data completeness (focusing on missing records rather than missing attribute values) and *temporal consistency* issues related to the tension between updating views as soon as new data arrive and waiting until all the expected data have arrived. Section 7 focuses on techniques for

characterizing errors, while Sect. 8 discusses correcting data quality problems. We discuss distributed data quality issues in Sect. 9, and we conclude in Sect. 10 with directions for future work.

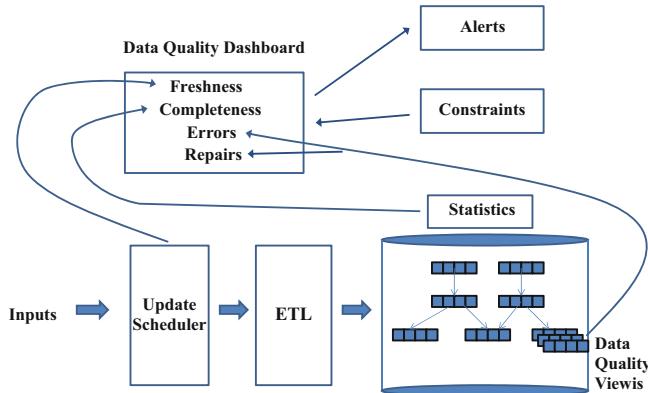
## 2 Quality-Aware Data Warehouse Design

It might appear that a solution to maintaining data warehouse quality is simple: to maximize data freshness (in the context of stream data warehousing), whenever new data arrive, we immediately update the corresponding base table and all the related materialized views; to ensure data correctness and consistency, we throw out or correct erroneous data during the ETL process. However, multiple feeds may generate new data at the same time, and running too many concurrent updates can be inefficient due to CPU, memory, and disk contention as well as frequent context switching [22]. Furthermore, some views may be more critical than others and should be given higher priority during updates. This motivates the first component of quality-aware data warehouse design: an *update scheduler* that decides which tables to update and when (more details in Sect. 3; a related notion of data currency will be discussed in Sect. 4).

Another reason why immediate update propagation is not appropriate is that data feeds are typically not synchronized; in fact, even the data within a single feed may arrive out of order. Thus, even though we want to maximize data freshness, we may have to delay propagating new data throughout the materialized view hierarchy in order to keep users from obtaining *incomplete* or *inconsistent* answers (more details in Sects. 5 and 6).

Now, to understand why detecting and correcting errors during the ETL process may not be feasible, note that while simple errors can be dealt with quickly during ETL, many problems and inconsistencies can only be detected after loading the data and correlating multiple data sources. In fact, some issues may only appear after the application has processed the data and users have noticed something suspicious when looking at the results. Even if we could find all the errors early, adding complex data quality processing to the ETL process will severely impact data freshness, as all the data, even clean data, could not be loaded until all the error checking is complete.

Furthermore, there is usually more than one way to correct a data quality problem, and it may not be clear what to do at the ETL stage (more details in Sect. 8). In fact, different corrections may make sense for different applications, and some applications may not even require completely clean data. However, if we choose a particular correction during ETL, we lose the original data and the ability to compute other possible corrections. Finally, an advantage of loading all the data into the warehouse, even data that are possibly incorrect, is that we can efficiently query and analyze erroneous data in order to understand the root causes of data quality problems (more details in Sect. 7).



**Fig. 2** An abstract architecture of a quality-aware data warehouse system

Motivated by these observations, Fig. 2 illustrates an abstract architecture of a data warehouse system augmented with features that characterize and ensure data quality. As mentioned earlier, the update scheduler determines the order in which base and derived tables are updated. Furthermore, a data quality dashboard reports the current state of the warehouse and helps understand data quality problems. There are at least four functionalities that a data quality dashboard should support, as detailed below.

The freshness component may report the time of the most recent update of each table; a more sophisticated method could be to query each table for the most recent record timestamp. One advantage of the former is that it is nonintrusive: it suffices to examine the scheduler logs without having to query the warehouse. Notably, the freshness report may be used to understand and debug the update scheduler. For example, if base tables are fresh but derived tables are out of date, then perhaps the scheduler parameters should be adjusted to update derived tables more often. Furthermore, a data quality monitoring component may consult the freshness report and send out alerts if important tables have not been updated for more than a specified amount of time.

Even if a table has recently been updated, it is not clear from a simple freshness report how many records were inserted or if all the expected data have arrived. For this purpose, the completeness module may report the number of records in each partition of every table; these may be obtained from statistics tables that store per-partition counts, histograms, or sketches. Again, an alert may be triggered if a certain partition has much more (or much less) data than, say, the previous partition. In some cases, it may also be possible to define specific completeness constraints. For instance, we may know that a data feed consists of four files every 5 min or 1,000 records every minute, in which case receiving fewer than four files or 1,000 records may trigger an alert (more details in Sect. 5).

The next two dashboard components need to examine the data in more detail. The error report may list, for each table (and each partition), the records believed to

be incorrect or those which cause a violation of data quality constraints (constraints may be supplied by the user or discovered from data [6,11,14]). We can keep track of this information by maintaining materialized views with record IDs and the errors or inconsistencies they cause. Other examples of error reports include measuring the percentage of records per partition that fail each of the specified constraints or summarizing the violating records (more details in Sect. 7). The repair component may suggest possible ways to correct the data in order to eliminate constraint violations (more details in Sect. 8). Two examples of data quality dashboards that focus on profiling and correcting errors are Data Auditor [24] and Semandaq [12].

We can leverage the view maintenance capabilities of a warehouse system to precompute information required by the error and repair components. Examples of data-quality-related views include errors lists (see, e.g., [13] for finding violations of conditional functional dependencies expressed as an SQL query whose output may be saved as a materialized view), possible corrections of the data (see, e.g., [4,9,34] in the context of functional and conditional functional dependencies), and samples that enable efficient estimation of the fraction of records that fail a given constraint (see [5] for warehousing sample data and [10] for sampling to estimate the fraction of violations with respect to a functional dependency).

To summarize this section, a quality-aware data warehouse requires scheduling of updates to maintain data freshness as well as a mechanism for reporting and generating alerts in case of problems with data completeness, correctness, and consistency. In some cases, we may be able to generate various fixes of the data and, optionally, use them to answer queries.

### 3 Data Freshness

A simple approach to updating a stream warehouse is to load new data into the corresponding base table upon arrival and then update the affected views. However, in practice, a warehouse system may have to handle hundreds or more sources and maintain hundreds or more materialized views. Running too many parallel updates may lead to resource contention, causing delayed updates and poor data freshness. Adding an update scheduler is a way to control resource usage and order the updates in a way that improves data freshness.

When formulating a scheduling problem, we require an optimization metric, such as minimizing the number of missed deadlines. In a data warehouse context, we could define the deadline of an update as the arrival time of the next update for the given table. However, this only works for regular data feeds, and it is not clear how to extend this definition to derived tables. Instead, a more intuitive metric is to maximize data freshness or, equivalently, minimize data staleness (summed or averaged over all the tables in the warehouse). There are many ways to define data staleness [1,2,8,37]; for example, we can take the difference between the current time and the timestamp of the most recent record loaded into a given table. If the warehouse includes tables with varying priorities, we should minimize the total priority-weighted staleness.

A simple greedy heuristic to minimize priority-weighted staleness is as follows [22]. At any point in time, we identify tables and materialized views that are out of date, i.e., this includes base tables that have received new data on their corresponding feeds or derived tables that were updated less recently than their source tables. Of all the tables that are currently out of date, we update the one whose priority-weighted staleness can be improved by the greatest amount per unit of processing time (assuming that we can estimate the update processing time).

The above strategy updates one table at a time regardless of view hierarchies. For example, suppose we have a base table  $B1$  and a view  $V1$  defined as a query against  $B1$  and another base table  $B2$  and a view  $V2$  defined as a query against  $B2$ . If new data arrive for  $B1$  and  $B2$  and we choose to update  $B1$  first, we may not necessarily update  $V1$  next; for example, we may choose  $B2$  instead if it is more stale or if it has a higher priority. Since preempting an update process is not straightforward (we would have to suspend the ETL process and remember how much data we have loaded), if a batch of new data arrives for another table with very high priority, that update will have to wait (at least) until the currently running update is finished.

Suppose that several batches of new data have arrived for a base table  $B$ . When  $B$  is scheduled for an update, we have a choice of loading some or all the new data into  $B$ . Since we eventually have to load all the data to enable historical queries, the simplest option is to load all the new batches. However, if the data warehouse system is overloaded and struggling to keep up, we may choose to load only the oldest batch and move on to updating another table, or only the youngest batch (and leave a “hole” in the table that will be patched in a future update). The latter is appropriate if  $B$  is used by a real-time application that can tolerate data loss but requires the most recent data at all times.

Note that updating one table at a time in the order chosen by the scheduler means that related tables (e.g., views computed from the same source tables, or a base table, and the derived tables computed from it) may have different freshness. One may consider this lack of synchronization to be a data quality problem for queries or analyses that join these related tables. A simple solution to this problem is to synchronize the data at query time: take the least fresh table that is being accessed—suppose this table has data up to 5 o’clock today—and, for the purposes of the query, ignore records from the other tables with timestamps larger than 5 o’clock.

## 4 Data Currency

Data freshness measures how well we can keep up with continuous data feeds. A stale table may still be considered “correct”; it just does not contain recent history. For example, a fact table may be stale because it only contains sales transactions up to one hour ago. Data currency is a related quality issue, which indicates whether the most recent data in a warehouse are still correct, or valid, as of the current time.

Dimension tables may be affected by data currency problems. For example, suppose that a customer moves to a different city but her address is not updated in

the customer dimension table. Any views that join the dimension table with the fact table of customer activity, with a group-by on city, will consider our recently moved customer’s activity to have taken place in her old city. These types of problems are difficult to detect—without an explicit update, we have no reason to suspect that a customer has moved, and we will continue using the current dimension attribute values. Often, the only hope is that when the updated address finally arrives, it also comes with a timestamp indicating when the move took place. If so, then we can revise (recent partitions of) views that may have incorrectly used the old address between the move date and the date the update arrived.

A related problem is to verify that an update to a dimension table “makes sense” via *currency constraints* that govern the evolution of data over time [16]. For example, a person’s marital status may be updated from never married to married, but not vice versa. These constraints are useful when updates to dimension tables arrive without timestamps, and it is not clear which is the current value.

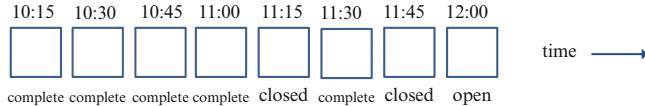
## 5 Data Completeness

Even if a base table is fresh and was updated recently, it is not clear if all the expected data have arrived. Unfortunately, missing data, delayed arrivals, and out-of-order arrivals are common in large warehouses that collect data from multiple uncontrolled sources [20, 33]. For example, some sources may be down and they may have stopped producing data or responding to requests for data; the data collecting mechanism may not have received all the data or has not yet been configured to start collecting data from newly added sources; some data may get lost or delayed on their way to the warehouse; and sources that were temporarily off-line may come back online and send a mix of new and older data. While some alerting or monitoring applications must process data on the fly and may find delayed data of little value, a data warehouse must load all the data to enable historical analyses.

In some cases, we can formulate constraints that define data completeness based on the semantics of data sources or the data collecting mechanism. For example, a data feed may be configured to send four files every 15 min, and we might know that files rarely arrive more than an hour late. Clearly, if we receive all four files, the data are complete. Moreover, if we only receive three files, but an hour has passed, then we can infer that no more data for the given 15-min window will arrive. Formalizing these observations, we can associate *completeness levels* with each temporal partition of a table [20]:

- A partition is *open* if some data are in it and may be added to it in the future.
- A partition is *closed* if we do not expect to receive any data for it in the future.
- A partition is *complete* if it is closed and all the expected data have arrived (no data are permanently lost).

The open and complete levels are the natural weakest and strongest definitions, with closed partitions representing “stable” data with permanently missing pieces. Other



**Fig. 3** An example of completeness markers

completeness levels such as half-closed or half-complete as well as record-level refinements of existing levels (such as complete for data arriving from American customers, open for data arriving from European customers) are also possible [20].

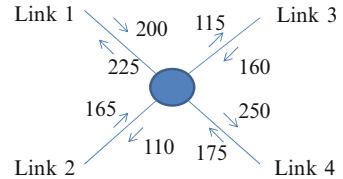
Figure 3 shows a table consisting of eight partitions, each storing 15 min of data; e.g., the oldest (leftmost) partition stores data whose timestamps are between 10:00 and 10:15. Each partition is labeled with its completeness level. A convenient way to summarize the state of this table (and a useful piece of information to display on a data quality dashboard) is to say that it is closed up to 11:45 (all the partitions up to that time are at least closed, perhaps complete). Note that complete partitions may not form a contiguous time interval since some partitions may have permanently missing data and will never become complete.

So far, we have discussed applying completeness constraints to base tables. We can apply the same framework to derived tables if we know which partitions of the source tables were used to compute a given partition of the derived table (this information is typically specified by the user or inferred by the system in order to enable incremental view maintenance [19, 21]). A derived table partition is closed (resp. complete) if all the partitions that were used to compute it are at least closed (resp. at least complete).

The above completeness constraints are useful if we can characterize how much data we expect to receive. However, some data sources may deliver varying amounts of data over time. For example, the number of system log messages or errors collected from an IP network or a data center depends on the state of the system. In these situations, we can still detect data completeness problems by comparing against the amount of data received in the past (e.g., at the same time yesterday, at the same time on the same day last week) using time series auto-regression models. Many data feeds exhibit obvious temporal and seasonal patterns, and deviations from these patterns may indicate data quality errors—or suspicious events happening in the system being monitored. Either way, these deviations are worth reporting and investigating further.

In some cases, detecting missing data requires more complex constraints. For example, suppose we are monitoring an infrastructure network—an IP network, a road network, or a power grid—consisting of nodes (routers, road intersections, power stations) and edges (links, roads, power lines). We may receive data feeds from routers or sensors that count the number of items (packets, cars, etc.) passing through each edge. Over time, we expect a *conservation law* [23, 35] to hold at each node in the network: the total number of packets entering a router on all the inbound links equals the total number of packets exiting on all the outbound links, the total number of cars entering an intersection from each direction equals the total number

**Fig. 4** A network node with four links and counts of incoming and outgoing traffic (adapted from [23])



of cars exiting the intersection, etc. The conservation law may not hold exactly at all times due to delays, but it should be true in the long run.

For example (adapted from [23]), consider Fig. 4, which illustrates a router with four bidirectional links connected to it (Link 1 through Link 4) as well as the incoming and outgoing traffic counts at some point in time. Although there may be more incoming than outgoing traffic on a single link (or vice versa), the total incoming traffic at the node level should equal the total outgoing traffic, as is the case in Fig. 4. Now suppose that we are not aware of Link 4; perhaps it has recently been connected and has not yet been added to the network topology table, and therefore it is not being monitored. This type of missing data is very hard to detect unless a network technician physically inspects the router and observes the new link connected to the router. However, applying the conservation law to this router without the traffic flowing on Link 4, we get 525 total incoming packets and 450 total outgoing packets, which is a violation of the law. Thus, by aggregating related measurements, we are able to detect that there may be an unmonitored link (or that measurements coming from existing links are wrong).

## 6 Temporal Consistency

Temporal consistency refers to producing correct answers to queries and correct updates to materialized views in the presence of unsynchronized and possibly incomplete data. Temporal consistency problems may occur in a stream warehouse that updates base tables and materialized views as soon as new data arrive. In particular, it may not be clear which views or queries contain “stable” results and which results may change over time. A warehouse system must provide temporal consistency guarantees so that users can interpret the results of queries against continuously arriving data.

Fortunately, we can use the completeness levels described in the previous section to characterize temporal consistency [20]. Consider a derived table  $D$  that computes daily aggregates over a base fact table  $B$  that is updated continuously throughout the day. Assume that  $B$  contains purchase transactions identified by a customer ID and a timestamp. Suppose that  $B$  is partitioned by hour and  $D$  is partitioned by day, i.e., computing a new partition of  $D$  requires aggregating 24 newest partitions of  $B$ . Since  $D$  computes daily statistics, it likely does not have to be updated throughout the day as new data arrive, only after all the data have arrived. This means that we

should create a new partition of  $D$  with data from the current day only when all 24 of the newest partitions of  $B$  are closed (i.e., after all the expected data from the current day have been loaded into  $B$ ). This way,  $D$  will maintain a closed completeness level, and queries against it are guaranteed to read sales totals that will not change in the future.

On the other hand, suppose that we want to maintain another derived table, call it  $D_2$ , that selects “suspicious” purchase transactions from  $B$  (e.g., expensive purchases, unusual purchases, multiple purchases made around the same time in different locations). Here, it makes sense for  $D_2$  to be loaded as often as  $B$  so that applications that use  $D_2$  can view and react to suspicious data as soon as possible. This means that  $D_2$  and  $B$  both have an open completeness level, and queries against recent partitions of  $D_2$  or  $B$  may give different answers at different points in time.

Similarly, suppose the warehouse also maintains a derived table  $D_3$  that selects customers who spend a total of at least 1,000 dollars in a given day.  $D_3$  may also be maintained with an open completeness level and may be updated continuously throughout the day. Again, queries may see different results at different points in time since new customers may be added to (the recent partitions of)  $D_3$  as soon as their daily spending total reaches 1,000 dollars. Clearly, customers already in  $D_3$  are guaranteed to stay in  $D_3$ , no matter what other records arrive during the rest of the day.

Note that the above consistency guarantees require a slight change to the update scheduler. When a table is due for an update, the scheduler must determine which partitions of the source tables are required to perform the update. If all of these partitions have the required completeness level, the update may proceed. Otherwise, the update should not yet be invoked at this time.

Also, note that the same temporal consistency mechanism works for ad hoc queries that contain timestamp predicates, e.g., an ad hoc query over the past week’s data from table  $T$ . It suffices to find all the partitions of  $T$  required to answer the query and examine their completeness level. For example, if all the required partitions are closed, then the answer to the query is guaranteed to be closed as well. Similarly, suppose that the query requires complete results; e.g., perhaps we are generating a business report that makes sense only if all the expected data are available. Then, this is possible only if all the partitions accessed by the query are also complete. In general, views and queries used by real-time monitoring or alerting applications can sacrifice temporal consistency for the sake of data freshness and can tolerate an open completeness level. Views and queries for off-line analytics tend to require closed or even complete data.

## 7 Detecting and Profiling Data Quality Problems

So far, we have examined data quality issues related to the temporal nature of a data warehouse, including missing, delayed, and incomplete data. We now turn to problems involving incorrect data.

## 7.1 Error Detection

Finding and tagging erroneous records or attribute values is the first step. A common error detection method is to formulate logical constraints or patterns that define the expected semantics of the data; violations of the expected semantics may indicate data quality problems. Simple constraints may be expressed as conditions on individual records; for example, we may assert that in an Employees table, Age and Salary must be nonnegative. Testing these predicates one record at a time suffices to find violations. Other types of constraints need to examine multiple tuples to detect violations. For example, due to the temporal nature of a data warehouse, we may want to validate constraints that refer to previously arrived tuples from the same source or customer; e.g., we may want to assert that two consecutive temperature readings from the same sensor should be similar. Similarly, functional dependencies require at least two conflicting tuples to cause a violation, so a newly arrived tuple must be compared against existing tuples for potential violating pairs.<sup>1</sup> Finally, some constraints such as inclusion dependencies are defined across multiple tables and may require joins to detect inconsistencies.

For numeric data, in addition to constraint-based error detection we may also employ statistical error detection techniques, such as distribution or distance-based outliers (see, e.g., [27] for a survey of quantitative data cleaning and [28] for a survey of outlier detection methodologies). These techniques may assign a probability of correctness to each record, whereas constraint-based approaches typically classify records as either satisfying or failing a given constraint.

## 7.2 Error Profiling and Summarization

Since the number of records that violate a constraint or a statistical model may be large, a data quality dashboard requires techniques to summarize the extent of data quality problems. A useful data quality metric is the percentage of records tagged as erroneous. Since warehouse tables are partitioned by time, it makes sense to compute these percentages separately for each partition.

For record-level predicates, we can simply compute the fraction of tuples for which the predicate fails. For functional dependencies, we can compute the number of violating pairs of tuples as a fraction of the total number of pairs of tuples or the maximal fraction of data that does not cause any violations [25, 32]. We will denote these fractions as the *confidence* of a constraint on a given table. For instance, a confidence of 90 % roughly means that 90 % of the data satisfy the constraint.

Data in a large warehouse tend to be very heterogeneous across different data sources and across time. Furthermore, some data feeds (such as sensor

---

<sup>1</sup>Recall that a functional dependency (FD)  $X \rightarrow Y$  asserts that two tuples having the same value of the left-hand-side attributes ( $X$ ) must also agree on the right-hand-side attributes ( $Y$ ).

**Table 1** A data quality view of router performance reports tagged with error information

Name	Location	Time	Error
Router1	New York	10:00	1
Router2	New York	10:00	1
Router3	Chicago	10:00	0
Router4	Chicago	10:00	0
Router1	New York	10:05	0
Router2	New York	10:05	1
Router3	Chicago	10:05	1
Router4	Chicago	10:05	0
Router1	New York	10:10	1
Router2	New York	10:10	1
router3	Chicago	10:10	0
Router4	Chicago	10:10	0
Router1	New York	10:15	1
Router2	New York	10:15	0
Router3	Chicago	10:15	1
Router4	Chicago	10:15	1

measurements or system logs) are machine generated and may be prone to systematic or correlated problems [3] or problems that affect some types of sources or measuring devices more than others. Thus, in addition to reporting the confidence of the entire table or partition, it is useful to identify subsets of the data that satisfy (i.e., have high confidence) or violate (i.e., have low confidence) a constraint. A set-cover-based summarization technique has recently been proposed that chooses a small number of semantically meaningful subsets to represent data having high or low confidence [24].

To illustrate this approach, consider a network monitoring data warehouse that collects performance data from routers and links in an IP network. Suppose that we have a fact table that stores periodic measurements from each router, such as the incoming and outgoing traffic and CPU usage. Also, suppose we have a dimension table that describes routers in terms of their IP address, location, model type, etc. Table 1 shows a fragment of a data quality view containing some of the dimension attributes (router name, location, and measurement time) and an “error” attribute denoting whether the corresponding performance measurement violated some constraint. The actual measurements are not included in the data quality view, only an indication of their correctness. The confidence of the router performance table (with respect to the given constraint) is  $\frac{7}{16}$  since only seven records are clean (their error attribute is zero).

We use *patterns* of values of dimension attributes to represent subsets of a relation; a summary will contain a selected set of patterns that concisely describe which portions of the data satisfy (respectively, violate) a given constraint. Let  $a_1, a_2, \dots, a_\ell$  be the dimension attributes. A pattern contains  $\ell$  symbols, one for each dimension attribute. Each symbol is either a value from the corresponding

**Table 2** A summary of subsets of the router performance table that violate a constraint

Name	Location	Time
-	New York	-
-	-	10:15
Router3	-	10:05

attribute's domain or a special "wildcard" symbol "-". Let  $p_i[a_j]$  denote the symbol corresponding to the  $j$ th dimension attribute of the  $i$ th pattern included in the summary, and let  $t[a_j]$  be the value of the  $j$ th attribute of a tuple  $t$ . A tuple  $t$  is said to *match* a pattern  $p_i$  if, for each  $a_j$  in  $A$ , either  $p_i[a_j] = \text{``-''}$  or  $t[a_j] = p_i[a_j]$ . We define  $\text{cover}(p_i)$  to be the set of tuples that match  $p_i$ . The confidence of a pattern  $p_i$  is defined as the confidence of a sub-relation containing only the tuples from  $\text{cover}(p_i)$ . Note that the pattern consisting of only the "-" symbols matches the entire table.

For example, consider the pattern  $p = (-, \text{ New York}, -)$ , i.e., all routers from New York, over the data quality view from Table 1. There are eight matching rows in the table, so  $|\text{cover}(p)| = 8$ , of which only two are correct. Thus, the confidence of this pattern with respect to the given constraint is  $\frac{2}{8}$ .

Intuitively, a good summary contains patterns that (satisfy the given confidence requirements and) are as general as possible in order to clearly capture the semantics of the data with respect to the given data quality constraint. Returning to Table 1, if all tuples from New York violate the constraint, then they should be summarized using a single pattern  $(-, \text{ New York}, -)$  rather than multiple patterns that refer to individual routers from New York.

The input to the summarization process consists of a data quality view with a binary "error" attribute denoting the satisfaction or violation of some constraint, an integer  $k$  denoting the maximum number of patterns that may be included in the summary, and a fraction  $\hat{c}$ , which is either a lower bound or an upper bound on the confidence of each pattern. Setting a lower bound on the confidence summarizes satisfying tuples, while an upper bound summarizing the violations. The output is a summary consisting of at most  $k$  patterns, each of confidence at least (respectively, at most)  $\hat{c}$ , that collectively cover the largest possible fraction of the data. This corresponds to the maximum set coverage problem (where the possible sets correspond to patterns that satisfy the specified confidence requirement), whose solution is NP-hard but can be effectively approximated using a simple greedy heuristic: at each step, choose the set (pattern) that covers the most uncovered tuples.

Table 2 shows a summary of the quality of the router measurement view from Table 1, given that  $k = 3$  and  $\hat{c} \leq 0.25$ , i.e., we are interested in subsets with at most 25 % of clean tuples (on average). This summary suggests that data coming from New York have a large number of violations, as well as data that arrived at 10:15 and data from a particular router that arrived at 10:05. Additionally, one can generate related summaries with different  $\hat{c}$  parameters to discover subsets with varying degrees of constraint violations. Alternatively, by changing  $\hat{c}$  to a lower

bound, say,  $\geq 0.9$ , we can identify “well-behaved” subsets that satisfy the constraint with high confidence. For example, the pattern  $(-, \text{ Chicago}, 10:00)$ , which has a confidence of one, might be included in such a summary.

Note that summarizing the behavior of a constraint in this fashion may be easier for users to interpret and more helpful in identifying the root causes of data quality problems, than a (possibly very long) list of individual violations. Also, note that we can summarize the satisfaction or violation of a constraint already specified on the warehouse, or we may “try out” various other constraints to see if they would hold on a given table or view. This type of exploratory data analysis can help understand data semantics and discover new types of data quality problems.

We conclude the discussion of error detection and profiling by pointing to the body of knowledge on data lineage and provenance. Knowing how, where, or why a particular data item was derived can help point out erroneous data sources and processes.

## 8 Correcting Data Quality Problems

In the previous section, we discussed identifying and summarizing data quality problems. In this section, we discuss correcting these problems.

Some organizations maintain a repository of master data [17], which may be used to correct errors even before the data are loaded to the warehouse. However, in general, there may be many ways to correct an error, and furthermore, different applications may prefer different corrections. A reasonable strategy is to save the original data and store the possible repairs as materialized views.

There are many statistical and machine learning approaches that are relevant to data cleaning. For example, suppose that a tuple with attributes  $A$ ,  $B$ , and  $C$  is missing the value of  $C$ . Using a classifier trained on clean data or an association rule mining algorithm, we can predict the missing value of  $C$  based on the value of  $A$  and  $B$ . Optionally, we can assign a probability that the predicted value is correct [31] (data uncertainty is covered in more detail in a separate chapter of this handbook).

Additionally, there are many domain-specific statistical cleaning strategies. For example, we can clean sensor data feeds by exploiting spatial and temporal correlations [29]. Suppose that a temperature sensor reports an unusually high or unusually low reading. We can delete this reading, change it to the value previously reported by this sensor (or the average of recent values), or set it to an average value reported by other temperature sensors located nearby.

There are also constraint-based cleaning techniques which modify the data to satisfy the given constraint. Figure 5 gives an example in the context of functional dependencies (FDs). At the top, we illustrate a table storing vehicle makes and models that does not satisfy the FD  $Model \rightarrow Make$ . In particular, tuples  $t_1$ ,  $t_2$ , and  $t_3$  violate the FD since they have the same Model but different Make. There are many ways to resolve FD violations; we show four possible corrections at the bottom of Fig. 5. The leftmost correction changes the Make attribute of  $t_1$  to be equal to that of  $t_2$  and  $t_3$ ; the next one modifies the Make of  $t_2$  and  $t_3$  to be the same

Original table

	Model	Make
$t_1$	Corolla	Toyota
$t_2$	Corolla	Ford
$t_3$	Corolla	Ford
$t_4$	Accord	Honda

	Model	Make
$t_1$	Corolla	Ford
$t_2$	Corolla	Ford
$t_3$	Corolla	Ford
$t_4$	Accord	Honda

	Model	Make
$t_1$	Corolla	Toyota
$t_2$	Corolla	BMW
$t_3$	Corolla	BMW
$t_4$	Accord	Honda

	Model	Make
$t_1$	Camry	Toyota
$t_2$	Corolla	Ford
$t_3$	Corolla	Ford
$t_4$	Accord	Honda

Possible corrections

**Fig. 5** An example of possible repairs of FD violations

as the Make of  $t_1$ ; the next modifies the Make attributes of all three violating tuples, while the last one changes the Model of  $t_1$  to be different to the Model of  $t_2$  and  $t_3$ . Since it is not always clear which repair is correct, a data warehouse may store multiple repairs for a given constraint (see, e.g., [4] for an algorithm that generates a sample of FD repairs that make nonredundant modifications to the data).

Finally, we note that there exist constraint-based cleaning approaches that modify both the data and the constraints in a nonredundant way so that the modified table does not violate the modified constraints (see [7] for an example of this approach in the context of functional dependencies). These approaches are useful in situations where the data semantics or business rules governing the data may have changed over time.

## 9 Distributed Data Quality

Large data warehouses are often distributed across many servers or data centers, making data quality analysis more difficult. For example, as we mentioned in Sect. 7, we may need to examine multiple tuples in order to detect constraint violations. If these tuples are on different servers, they may have to be shipped to a single location for analysis. In general, when designing distributed data quality algorithms, we want to minimize communication costs and/or balance the load across processing nodes.

For example (adapted from [15, 18]), consider finding tuples that violate a functional dependency in a distributed setting. Table 3 illustrates a (very simple) data set with three attributes ( $A$ ,  $B$ , and  $C$ ). Assume this data set is distributed across three servers as indicated in the server column; we have also labeled each tuple with a tuple ID.

**Table 3** Example of a table partitioned across three servers

Server	Tuple ID	A	B	C
1	t1	a1	b1	c1
1	t2	a2	b2	c2
1	t3	a2	b3	c3
2	t4	a1	b4	c4
2	t5	a1	b1	c1
2	t6	a1	b1	c5
2	t7	a2	b2	c6
3	t8	a1	b1	c1

Suppose that we want to detect the violations of the FD  $AB \rightarrow C$ . First, notice that tuple  $t_6$  conflicts with  $t_1$ ,  $t_5$ , and  $t_8$ ; also,  $t_2$  conflicts with  $t_7$ . However, we cannot detect these conflicts locally at each server. A simple strategy may be to choose a single server to which we will ship all the tuples from this particular table that are stored at the other servers. To minimize communication, we should choose server 2 since it stores the largest fragment of the table. This results in four tuples being shipped (if we had chosen server 1, we would have to ship five tuples to it; if server 3, we would ship seven tuples).

An improved solution is based on the observation that only tuples that have the same values of  $A$  and  $B$  can conflict with each other. This means that we can group the table on  $A$  and/or  $B$  and independently check for conflicts in each group, possibly on different servers [15]. Returning to Table 3, we may decide to ship tuples with  $A = a1$  to server 2 (because it stores three such tuples, whereas the other servers only store one each) and tuples with  $A = a2$  to server 1 (which stores two such tuples, whereas server 2 stores only one and server 3 stores none). That is, we ship tuples  $t_1$  and  $t_8$  to server 2 and  $t_7$  to server 1. Now server 2 can detect that  $t_1$ ,  $t_5$ ,  $t_6$ , and  $t_8$  conflict with each other, while server 1 has enough information to detect that  $t_2$  conflicts with  $t_7$ . Compared to the previous solution of validating the FD on server 2, we only have to ship three tuples, not four. Furthermore, we have distributed the error-checking load across two servers instead of doing all the work on one server.

## 10 Conclusions and Future Work

In this chapter, we discussed data quality problems and solutions in the context of data warehouses. We explained the unique features of data warehouse systems that lead to novel data quality concerns, including the temporal, distributed, and sometimes machine-generated nature of data, and the real-time nature of updates to base tables and materialized views. We then described solutions for data quality problems arising from these features, both from a system design and from an algorithmic standpoint, including update scheduling to maximize data freshness, ensuring data currency and completeness, balancing data freshness and temporal

data consistency, detecting, summarizing and correcting errors, and distributed detection of constraint violations.

There are many interesting topics for future work in data warehouse quality. In the context of stream data warehousing, there is an increasing need to make data-driven decisions in nearly real time, which requires further work on reducing data latency and improving freshness. Furthermore, new *incremental* data cleaning algorithms are required. In general, as new data sources become available (e.g., social media streams and energy usage measurements from smart grids), data warehouse systems will require new types of constraints to model the semantics of these sources and detect inconsistencies. Another interesting topic is to reduce the volume of data quality alerts. For example, if a base table and all materialized views sourced from it are experiencing a problem, then perhaps it suffices to send out one alert indicating a problem originating at the base table. Finally, given the popularity of the MapReduce framework for scalable data analytics, another direction for future work is to express data quality operations, such as error detection and data repair, as map-reduce jobs.

## References

1. Adelberg B, Garcia-Molina H, Kao B (1995) Applying update streams in a soft real-time database system. In: SIGMOD conference, pp 245–256
2. Baer A, Golab L (2012) Towards benchmarking stream data warehouses. In: DOLAP
3. Berti-Equille L, Dasu T, Srivastava D (2011) Discovery of complex glitch patterns: a novel approach to quantitative data cleaning. In: Proceedings of the ICDE
4. Beskales G, Ilyas IF, Golab L (2010) Sampling the repairs of functional dependency violations under hard constraints. PVLDB 3(1):197–207
5. Brown PG, Haas PJ (2006) Techniques for warehousing of sample data. In: Proceedings of the ICDE
6. Chiang F, Miller RJ (2008) Discovering data quality rules. PVLDB 1(1):1166–1177
7. Chiang F, Miller RJ (2011) A unified model for data and constraint repair. In: Proceedings of the ICDE
8. Cho J, Garcia-Molina H (2000) Synchronizing a database to improve freshness. In: SIGMOD conference, pp 117–128
9. Cong G, Fan W, Geerts F, Jia X, Ma S (2007) Improving data quality: consistency and accuracy. In: VLDB, pp 315–326
10. Cormode G, Golab L, Korn F, McGregor A, Srivastava D, Zhang X (2009) Estimating the confidence of conditional functional dependencies. In: SIGMOD conference, pp 469–482
11. De Marchi F, Lopes S, Petit J-M (2009) Unary and n-ary inclusion dependency discovery in relational databases. J Intell Inf Syst 32(1):53–73
12. Fan W, Geerts F, Jia X (2008) Semandaq: a data quality system based on conditional functional dependencies. PVLDB 1(2):1460–1463
13. Fan W, Geerts F, Jia X, Kementsietsidis A (2008) Conditional functional dependencies for capturing data inconsistencies. ACM Trans Database Syst 33(2):1–48
14. Fan W, Geerts F, Li J, Xiong M (2011) Discovering conditional functional dependencies. IEEE Trans Knowl Data Eng 23(5):683–698
15. Fan W, Geerts F, Ma S, Müller H: Detecting inconsistencies in distributed data. In: Proceedings of the ICDE
16. Fan W, Geerts F, Wijsen J (2011) Determining the currency of data. In: PODS, pp 71–82

17. Fan W, Li J, Ma S, Tang N, Yu W (2012) Towards certain fixes with editing rules and master data. VLDB J 21(2):213–238
18. Fan W, Li J, Tang N, Yu W (2012) Incremental detection of inconsistencies in distributed data. In: Proceedings of the ICDE
19. Folkert N, Gupta A, Witkowski A, Subramanian S, Bellamkonda S, Shankar S, Bozkaya T, Sheng L (2005) Optimizing refresh of a set of materialized views. In: VLDB, pp 1043–1054
20. Golab L, Johnson T (2011) Consistency in a stream warehouse. In: CIDR, pp 114–122
21. Golab L, Johnson T, Spencer Seidel J, Shkapenyuk V (2009) Stream warehousing with DataDepot. In: SIGMOD conference, pp 847–854
22. Golab L, Johnson T, Shkapenyuk V (2012) Scalable scheduling of updates in streaming data warehouses. IEEE Trans Knowl Data Eng 24(6):1092–1105
23. Golab L, Karloff HJ, Korn F, Saha B, Srivastava D (2012) Discovering conservation rules. In: Proceedings of the ICDE
24. Golab L, Karloff HJ, Korn F, Srivastava D (2010) Data auditor: exploring data quality and semantics using pattern tableaux. PVLDB 3(2):1641–1644
25. Golab L, Karloff HJ, Korn F, Srivastava D, Yu B (2008) On generating near-optimal tableaux for conditional functional dependencies. PVLDB 1(1):376–390
26. Golab L, Tamer Ozu M (2010) Data stream management. Synthesis lectures on data management. Morgan & Claypool Publishers, San Rafael
27. Hellerstein JM (2009) Quantitative data cleaning for large databases. Keynote at QDB (technical report at db.cs.berkeley.edu/jmh/papers/cleaning-unece.pdf)
28. Hodge V, Austin J (2004) A survey of outlier detection methodologies. Artif Intell Rev 22(2):85–126
29. Jeffery SR, Alonso G, Franklin MJ, Hong W, Widom J (2006) A pipelined framework for online cleaning of sensor data streams. In: Proceedings of the ICDE
30. Jensen CS, Pedersen TB, Thomsen C (2010) Multidimensional databases and data warehousing. Synthesis lectures on data management. Morgan & Claypool Publishers, San Rafael
31. Khousainova N, Balazinska M, Suciu D (2006) Towards correcting input data errors probabilistically using integrity constraints. In: MobiDE, pp 43–50
32. Kivinen J, Mannila H (1995) Approximate inference of functional dependencies from relations. Theor Comput Sci 149(1):129–149
33. Krishnamurthy S, Franklin MJ, Davis J, Farina D, Golovko P, Li A, Thombre N (2010) Continuous analytics over discontinuous streams. In: SIGMOD conference, pp 1081–1092
34. Kolahi S, Lakshmanan LVS (2009) On approximating optimum repairs for functional dependency violations. In: ICDT, pp 53–62
35. Korn F, Muthukrishnan S, Zhu Y (2003) Checks and balances: monitoring data quality problems in network traffic databases. In: VLDB, pp 536–547
36. Labio W, Yerneni R, Garcia-Molina H (1999) Shrinking the warehouse update window. In: SIGMOD conference, pp 383–394
37. Labrinidis A, Roussopoulos N (2001) Update propagation strategies for improving the quality of data on the web. In: VLDB, pp 391–400

# Using Semantic Web Technologies for Data Quality Management

Christian Fürber and Martin Hepp

**Abstract** In the past decade, the World Wide Web has started to evolve from a Web that provides human-interpretable information to a Semantic Web that provides data that can also be processed by machines. With this evolution, several useful technologies for knowledge capturing, representation, and processing have been developed which can be used in large-scale environments. Semantic Web technologies may be able to shift current data quality management technology to the next level. In this chapter, we discuss how Semantic Web technologies can be employed to improve information quality. In particular, we outline their application for (1) data requirements and metadata management, (2) data quality monitoring, (3) data quality assessment, (4) validation of data entries, (5) as reference data, and (6) in the area of content integration.

## 1 Introduction

In the past two decades, the World Wide Web (WWW) has significantly improved the accessibility and integration of information with technologies such as uniform resource identifiers (URIs) and hyperlinks based on a client/server architecture. Recently, the WWW has started to develop from a “Web of Documents” to a “Web of Data.” In the “Web of Documents” machines can only interpret how data shall be rendered. In the “Web of Data” (also known as the Semantic Web) machines can also process the semantics of the published data. The Semantic Web thereby makes use of (1) knowledge representation techniques such as ontologies to represent data in a machine-readable way and (2) knowledge processing techniques such as querying and reasoning to further exploit the represented data (cf. [1]). In this chapter,

---

C. Fürber (✉) · M. Hepp

Universität der Bundeswehr München, E-Business and Web Science Research Group,  
Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany

e-mail: [c.fuerber@unibw.de](mailto:c.fuerber@unibw.de); [mhepp@computer.org](mailto:mhepp@computer.org)

we discuss strengths and weaknesses of these techniques and outline how they can be employed in the domain of data quality management. For a better understanding, we first provide a general outline about the specialties of data representation in the Semantic Web, potential contributions of Semantic Web technologies, and the biggest challenges of data quality management that may be addressed by Semantic Web technologies.

## ***1.1 Data Representation in the Semantic Web***

Data in the Semantic Web is usually represented in the Resource Description Framework (RDF) [2]. RDF is a knowledge representation model that arranges data in triples, which allow the formulation of statements about a resource in a subject, predicate, object structure. For example, with RDF we can express statements such as `Germany isPartOf Europe`. The combination of a subject (`Germany`) and an object (`Europe`) with help of a predicate relationship (`isPartOf`) is called a triple. The resource which is described by the triple is thereby always in the subject position. Hence, it is possible to make statements about resources in a structured, machine-interpretable way.

Statements about resources can also be used to classify resources, to define disjointness axioms, or to restrict class memberships. RDF, RDF Schema (RDFS), and the Web Ontology Language (OWL) provide several constructs for these purposes. Thus, it is possible to represent and publish complex knowledge in large semantic networks. These semantic networks are also known as ontologies and can be defined as a formal conceptualization of a domain of interest (cf. [3, 4]). Following the open principles of the WWW, ontologies may thereby be defined and published by anyone. Hence, it is possible to define and share knowledge of a specific domain in a common vocabulary.

## ***1.2 Potential Contributions of Semantic Web Technologies***

Semantic Web technologies combine Web architecture with techniques originating from information management, knowledge management, and artificial intelligence. Therefore, Semantic Web technologies may provide solutions to the following problems:

*Content Integration:* Other than local databases architectures, the Semantic Web allows the integration of content from distributed data and documents. Ontologies can be modeled independently from a specific information provider to structure information entities and relationships among them. Other than database schemata, ontologies are easily extendable without harming processing capabilities. Moreover, ontologies can be used to query or retrieve data from relational data sources [5] or

to structure unstructured information such as text.<sup>1</sup> Thus, ontologies may serve as a central point to retrieve and integrate data from structured and unstructured sources (cf. [6, 7]). Moreover, different ontologies can be interlinked, e.g., by defining equivalence relationships between entities with identical semantics, which is useful for heterogeneous information that has evolved over time. With the help of uniform resource identifiers (URIs), it is possible to uniquely name and identify things at Web scale and navigate to an information source with a single mouse-click. Thus, definitions of classes and properties can easily be looked up on the Web, and the provenance of data can be preserved via URIs when publishing data on the Semantic Web. Moreover, paths of related content can be discovered dynamically by data consumers and processing algorithms.

*Semantic Definition of Data:* Often data concepts suffer from misinterpretation, especially when using the same data for multiple different purposes. Misinterpretations of data may be drawn back to the limited capabilities of conventional information systems to model and publish the semantics of data. Semantic Web technologies may improve this area in at least two different ways. First, the expressivity of ontologies modeled with RDF and OWL facilitates a precise definition of data concepts. The degree of formality for the definition of concepts thereby varies from highly formal axioms to informal textual descriptions [4]. Second, it is best practice in Semantic Web architectures to provide at least a textual definition to all classes and properties of an ontology that can be looked up via URLs. Thus, Semantic Web technologies may help to improve the semantic definition of data due to explicit publication and modeling of data semantics.

*Transparency and Reuse of Quality-Relevant Knowledge:* Data quality management requires the collection and management of quality-relevant knowledge. In order to reduce manual effort, it is important to document, save, share, and process captured knowledge. Semantic Web technologies facilitate the representation and usage of quality-relevant knowledge. The captured knowledge may thereby be represented in RDF that can be processed by machines, rendered for human readability, or shared across the Web. This can save manual effort by explicitly representing knowledge in a sharable way that would usually be hidden in local databases, programming code, or human minds. Moreover, it may help to raise transparency about the assumed quality perception when performing data quality management tasks.

*Creation of a Shared Understanding:* The explicit representation of a domain in form of an ontology may point communication among stakeholders towards precise semantic distinctions among entities and relationships. Hence, the creation and use of ontologies may help to raise the level of understanding about data semantics and, therefore, reduce the amount of data quality problems due to misinterpretation.

*Deduction of New Knowledge:* The explicit representation of entities and relationships within an ontology facilitates automated reasoning of implicit knowledge. For example, if we define the triples “Jennifer isA Woman” and “Woman isA Human”, then a reasoner can derive that “Jennifer isA Human”.

---

<sup>1</sup>For example, with technologies such as OpenCalais, <http://www.opencalais.com>.

This capability of Semantic Web technologies may reduce the amount of explicitly represented information to a minimum and, therefore, may lead to a higher degree of consistency. Moreover, the wisdom of data may be exploited with help of reasoning, e.g., to identify inconsistencies among data requirements.

*Locality of Data Quality Management Solutions:* At present, many data quality tools, such as Talend Open Studio for Data Quality,<sup>2</sup> store their quality-related information, e.g., the metrics used to identify deficient data, locally in the client software. This hinders collaboration and the generation of consensual agreement about requirements. The architecture of the WWW facilitates direct access to information sources at a very large scale. Users can work in the same space to publish and consume information. Hence, the architecture of the Web facilitates collaboration within and across the boundaries of organizations which may help to efficiently manage the quality of data.

## 2 Big Challenges of Data Quality Management

Data quality management is a knowledge-intensive endeavor with several challenges on the way to achieve a high level of data quality. In order to understand the core challenges of data quality management, we need to take a closer look upon the definition of data quality and the role of data requirements in the data quality management process. We thereby see data as “materialized information.” Therefore, we do not differentiate between data quality and information quality.

### 2.1 A Philosophical View on Data and Information Quality

The terms “data quality” and “information quality” have been defined in many different ways as illustrated in Table 1. Although the listed definitions are different, they contain a common assumption: The quality of data can be determined by the comparison of data’s current state (status quo) to its desired state. In the definitions of Table 1, the desired state is defined by (1) consumer expectations (cf. [8–10]), (2) specifications (cf. [10]), and (3) task requirements (cf. [9, 11]).

From a simplified perspective, data quality is achieved when the current state of data meets the desired state of data. The desired state of data is thereby defined by data requirements, i.e., a “need or expectation that is stated, generally implied or obligatory” [12] for data. So from this simplistic view, we define data quality as the degree to which data fulfills requirements (cf. [12]). Furthermore, we believe that a generic definition of the term “data quality” must not constrain the authority that is eligible to define data requirements. Hence, in general, anyone is able to define data requirements, not only data consumers.

---

<sup>2</sup><http://www.talend.com> (last accessed on July 1, 2012).

**Table 1** Definitions of data quality and information quality

Authors	Data quality definition
Wang and Strong [8]	“[...] data that are fit for use by data consumers”
Kahn et al. [10]	“conformance to specifications” and “meeting or exceeding consumer expectations”
Redman [9]	“Data are of high quality if they are fit for their intended uses in operations, decision making, and planning. Data are fit for use if they are free of defects and possess desired features”
Olson [11]	“[...] data has quality if it satisfies the requirements of its intended use”

**Fig. 1** Data quality management process (cf. [13, 14])

## 2.2 *The Role of Data Requirements for Data Quality Management*

The core data quality management (DQM) process contains the phases “definition,” “measurement,” “analysis,” and “improvement” (cf. [13, 14]). As illustrated in Fig. 1, the phases of DQM are usually organized in a cycle, since DQM is a continuous process that should never stop as data and its requirements are subject to change.

The actual activities during a DQM process vary according to the chosen measurement method that measures the current state of data quality. In DQM, there are at least three different measurement methods: (1) surveys, (2) inventory taking, and (3) data analysis. Survey-based techniques aim to determine quality requirements and the level of data quality typically by asking data stakeholders directly (cf. [8]). Although it is reasonable to ask the people that have to deal with the data about their data quality perception, their subjective judgment may not be precise enough to determine the true level of data quality for two reasons: (1) data stakeholders may be driven by different interests than achieving a high level of data quality and (2) data stakeholders may not always account for non-individual data quality requirements imposed by business policies or laws. During an inventory taking, real-world objects are compared to the data that shall represent the object.

In cases where the data does not match the characteristics of the real-world object, the data is assumed to be deficient (cf. [15]). Inventory taking is only suitable for quality management of data that represents observable characteristics of objects. Moreover, inventory taking can be very laborious. For example, imagine comparing all your address data with the real locations. A more objective and less laborious measurement can be achieved via data analysis. In the following, we focus on this latter option and explain the data quality management process based on data analysis techniques with special regard to the role of data requirements.

During the definition phase one has to define the characteristics of high-quality data, which is usually bound to a specific perspective. This phase is probably the most critical in DQM since it defines the state that data shall obtain. Therefore, all other phases will build upon the definition of what constitutes high-quality data. The definition phase can thereby be conducted via the formulation of data requirements. For example, one could define that all products in the information system must contain information about its length, breadth, and depth. However, data requirements may be heterogeneous or even inconsistent with each other. For example, John considers the values “male” and “female” to be legal for the attribute “gender” in information system (IS) A, while Mary considers only the values “m” and “f” as legal for the same attribute in the same IS. In order to create a common picture about the quality problems of a data source, we have to define an unambiguous desired state of data. Therefore, we have three basic options in our example:

- Take over the perspective of John as desired state of the values for the attribute “gender” in IS A.
- Take over the perspective of Mary as desired state of the values for the attribute “gender” in IS A.
- Harmonize and combine the perspectives of John and Mary to a common desired state of the values for the attribute “gender” in IS A.

Whatever option is best, the example shall stress the importance of the transparent and comparable formulation and representation of data requirements for data quality management. Semantic Web technologies may help to create a common understanding about the characteristics of high-quality data.

During the measurement phase, the data requirements are used to identify data quality problems and evaluate the level of data quality of a data source. The results of the measurement phase typically provide the foundation for the analysis phase. The analysis phase aims at identifying the root cause of the data quality problem. It must be stressed that data quality problems can also be caused by incomplete or inconsistent data requirements. In these cases, the data requirement has to be corrected rather than the data. However, once the root causes of the data quality problem have been identified, the most suitable options for improvement have to be chosen and executed during the improvement phase. In many cases, data cleansing will not be the only required improvement action. Often business processes and interfaces between IS's have to be adjusted to effectively remove the root causes of the data quality problem (cf. [13]).

**Table 2** Generic data requirement types derived from data quality problem types with appropriate quality dimension

Data requirement	Data quality problem type	Dimension
Mandatory value requirements	Missing values Conditional missing values	Completeness
Syntactic requirements	Syntax violations Misspelling/mistyping errors Embedded values Imprecise values	Representational consistency
Legal value requirements	Syntax violations Misspelling/mistyping errors Embedded values Imprecise values False values Meaningless values Misfielded values	Representational consistency
Legal value range requirements	Out of range values Meaningless values False values	Accuracy
Illegal value requirements	False values Meaningless values	Accuracy
Functional dependency requirements	False values Referential integrity violations Incorrect references Contradictory relationships	Accuracy
Unique value requirements	Unique value violations	Accuracy
Duplicate instance identification requirements	Inconsistent duplicates Approximate duplicates	Not precisely assignable
Update requirements	Outdated values	Timeliness
Expiration requirements	Outdated values	Timeliness

### 2.3 Generic Data Requirement Typology

Data requirements are often stated imprecisely and without a clear reference object. For example, data consumers may say “the data must be accessible,” “the data must be timely,” or “the data must be accurate.” Most of these quality requirement statements lack sufficient detail to assess the quality of data. For example, the assessment of timeliness requires a precise definition of what timeliness means to the requirement owner.

In order to provide more guidance for the expression of data requirements, we have analyzed data quality problem types typical for relational databases [6, 16–18] to identify a set of generic data requirement types. This procedure assumes that problems are always caused when requirements are not fulfilled [19]. Table 2 illustrates the identified data requirement types with the respective data quality problem types that can be identified via each data requirement. Moreover, we mapped quality dimensions as identified by Wang and Strong [8] to the requirement

types as a suggestion for the configuration of quality assessment frameworks. The mapping is based on the definitions of the data quality dimension [8] and the potential influence of the data requirement on the quality dimension.

It must be stressed that some of the generic requirements are similar to the generic integrity rules from Codd “which define the set of consistent database states [...]” [20]. Other than Codd’s database integrity rules [21], we focus on the IS independent modeling of data requirements from a non-programmer perspective since we believe that many of the data requirements can be derived from business knowledge and directives.

The list of generic data requirements refers to data on an instance level. Besides the illustrated requirement types, there may be additional requirement types referring to other artifacts such as the data model or elements of the IS which may also influence the data quality perception of users.

### 3 Employing Semantic Web Technologies for Data Quality Management

At present, there are only few approaches which make use of Semantic Web technologies for data quality management. During our research, we discovered five major uses of Semantic Web technologies in the area of data quality management, namely:

- Collaborative representation and use of quality-relevant knowledge
- Automated identification of conflicting data requirements
- Semantic definition of data
- Use of Semantic Web data as reference data
- Content integration

In the following, we explain the use of Semantic Web technologies in these areas by discussing existing approaches.

#### 3.1 *Collaborative Representation and Use of Quality-Relevant Knowledge*

Data quality management is a knowledge-intensive discipline. As explained in Section 2.2, data requirements are quality-relevant knowledge, which defines the characteristics of high-quality data. This knowledge can be further processed to identify data quality problems and create quality scores for data quality dimensions such as completeness or accuracy. However, technologies that are currently in use for data quality management often miss appropriate mechanisms for the structured representation of data requirements. Commercial data quality tools often

provide formula editors that generate code snippets that represent processable data requirements. These code snippets are usually not portable to tools outside of the proprietary software and hard to create and interpret by business experts. Moreover, it is hard to check consistency among data requirements when using such code generators. Semantic Web technologies facilitate the representation of data requirements in a structured and sharable way and enable business experts to express data requirements in a format that is immediately processable by machines.

Until today there is no common standard to represent quality-relevant knowledge in Semantic Web environments. However, there are several vocabularies that allow the representation of quality-relevant information such as provenance information, data requirements, review scores, and data quality analysis results. An early approach of Bizer and Cyganiak [22] introduced the Web Information Quality Assessment framework (WIQA).<sup>3</sup> WIQA supports information filtering of high-quality information based on policies stated by data consumers and quality-related information published in the Semantic Web. The filtering policies are expressed in the framework-specific WIQA Policy Language (WIQA-PL). To filter information that meets the quality requirements of data consumers, WIQA processes context-, content-, and rating-based information represented in RDF graphs and structured via standardized Semantic Web vocabularies. Another approach from Fürber and Hepp provides a data quality constraints library<sup>4</sup> that contains rule templates for the representation of generic data requirements (also called data quality rules) based on the SPARQL Inferencing Notation (SPIN) framework<sup>5</sup> [23]. SPIN is a Semantic Web vocabulary and processing framework that facilitates the representation of rules based on the syntax of the SPARQL Protocol And RDF Query Language (SPARQL).<sup>6</sup> The data quality constraints library enables business experts to define data requirements for their data based on forms as part of the data quality management process. The SPIN framework then automatically identifies requirement violations in data instances. The operationalized data requirements are thereby represented in RDF files based on the SPIN vocabulary. This materialization of data requirements in a common vocabulary facilitates sharing of data requirements at Web scale and, therefore, raises transparency about the assumptions made when monitoring and assessing data quality. This SPIN-based data quality management approach has been extended to assess the quality state of data for the dimensions of semantic accuracy, syntactic accuracy, completeness, timeliness, and uniqueness [24]. Moreover, Brüggemann and Aden proposed a way to represent dependencies between attribute values within an ontology for further data quality management-related processing [25].

---

<sup>3</sup><http://www4.wiwiss.fu-berlin.de/bizer/wiqa/>.

<sup>4</sup><http://semwebquality.org/ontologies/dq-constraints#>.

<sup>5</sup><http://spinrdf.org/>.

<sup>6</sup>SPARQL is a query language for Semantic Web data similar to the Structured Query Language (SQL) for relational data; see specification at <http://www.w3.org/TR/rdf-sparql-query/>.

```

foo:PropertyCompletenessRule_1
  a      dqm:PropertyCompletenessRule ;
  dqm:testedClass http://www.example.org/MyClass ;
  dqm:testedProperty1 http://www.example.org/MyProperty ;
  dqm:requiredProperty "true"^^xsd:boolean ;
  dqm:requiredValue "true"^^xsd:boolean .

```

**Fig. 2** Example representation of a complete requirement with the DQM-Vocabulary

Recently, an ontology for data requirements management, data quality monitoring, and data quality assessment has been provided, called the Data Quality Management Vocabulary<sup>7</sup> (DQM-Vocabulary) [26]. The DQM-Vocabulary can be used to collect, represent, and manage data requirements. Similar to the SPIN-based approach, it facilitates the expression of data requirements in a common vocabulary. The DQM-Vocabulary is based on the generic data requirement types as listed in Table 2 and, therefore, covers a nearly complete set of possible data requirements on instance level. Figure 2 shows an example representation of a completeness requirement in Turtle syntax<sup>8</sup> for the values of the example property <http://www.example.org/MyProperty>. It defines in a machine-readable way that the values for this property cannot be missing. Processing data requirements via computer algorithms makes sense for multiple purposes:

1. Data can be monitored for requirement violations via comparison of the data requirement to the data behind the classes and properties specified in the data requirement.
2. Based on the detected requirement violations dimensional quality scores can be computed to represent the quality state of an information source.
3. Data requirements can be compared to each other to automatically identify inconsistent requirements.

In Semantic Web architectures, there are at least three basic ways to process RDF data, namely, via (1) reasoners, (2) Semantic Web programming frameworks, or (3) SPARQL queries. Reasoners are programs that use the represented logic of ontologies and/or user-defined rules (1) to infer implicit knowledge and (2) to check the logical consistency at ontology and instance levels [27]. Semantic Web programming frameworks, such as the Java framework Jena<sup>9</sup> or the Python framework RDFLib,<sup>10</sup> provide standard programming constructs that can be used to build your own processing applications for Semantic Web data. However, the easiest way to process Semantic Web data are SPARQL queries.

Similar to SQL, SPARQL queries provide capabilities to create reports or update data in Semantic Web environments. Based on data requirements that are expressed

<sup>7</sup><http://semwebquality.org/dqm-vocabulary/v1/dqm>.

<sup>8</sup><http://www.w3.org/2007/02/turtle/primer/>.

<sup>9</sup><http://jena.apache.org/>.

<sup>10</sup><http://www.rdf4j.net/>.

```

SELECT ?dqr ?i
WHERE {
  ?dqr a dqm:PropertyCompletenessRule .
  ?dqr dqm:testedClass ?tclass .
  ?dqr dqm:testedProperty1 ?tprop .
  ?dqr dqm:requiredValue "true"^^xsd:boolean .
  ?dqr dqm:requiredProperty "true"^^xsd:boolean .
  {
    ?i a ?tclass .
    FILTER NOT EXISTS{
      ?i ?tprop ?value .
    }
  }UNION{
    ?i a ?tclass .
    ?i ?tprop "" .
  }
}

```

**Fig. 3** Example of a data quality monitoring report with SPARQL

```

SELECT ?dqr ?tclassURI ?tpropURI (COUNT(?s) AS ?violations) (COUNT(?s2) AS ?total)
((?total - ?violations)/?total) AS ?completeness
WHERE {
  {
    ?dqr a dqm:PropertyCompletenessRule .
    ?dqr dqm:testedClass ?tclass .
    ?dqr dqm:testedProperty1 ?tprop .
    ?dqr dqm:requiredValue "true"^^xsd:boolean .
    ?dqr dqm:requiredProperty "true"^^xsd:boolean .
    ?tclass dqm:hasURI ?tclassURI .
    ?tprop dqm:hasURI ?tpropURI .
    BIND(IRI(str(?tclassreal)) AS ?tclassURI) .
    {
      ?s a ?tclassURI .
      FILTER NOT EXISTS{
        ?s ?tpropURI ?value .
      }
    }UNION{
      ?s a ?tclassURI .
      ?s ?tpropURI "" .
    }
  }UNION{
    ?dqr a dqm:PropertyCompletenessRule .
    ?dqr dqm:testedClass ?tclass .
    ?dqr dqm:testedProperty1 ?tprop .
    ?dqr dqm:requiredValue "true"^^xsd:boolean .
    ?dqr dqm:requiredProperty "true"^^xsd:boolean .
    ?tclass dqm:hasURI ?tclassURI .
    ?tprop dqm:hasURI ?tpropURI .
    ?s2 a ?tclassURI .
  }
}GROUP BY ?dqr ?tclassURI ?tpropURI

```

**Fig. 4** Example of a data quality assessment report with SPARQL

via an ontology such as the DQM-Vocabulary, it is, therefore, possible to create reports about data instances with requirement violations and reports with dimensional data quality scores. Figure 3 shows a SPARQL query that retrieves instances with missing properties and values. As prerequisite for the query represented in Fig. 3, it is necessary to define which properties need to have properties and property values for each instance, e.g., as shown in Fig. 2. The query shown in Fig. 3 is generic and processes all property completeness rules represented in the terminology of the DQM-Vocabulary. It results in a list with instances that violate the property completeness rules, which is useful for monitoring the satisfaction of data requirements. Due to the generic design of the query, a lot of manual work can be saved since the query logic only has to be defined once.

Moreover, it is possible to compute data quality scores for completeness based on the same data requirement as shown in Fig. 4. The query shown in Fig. 4 can

The screenshot shows a software interface for creating a 'Property Requirement'. The main title is 'Create PropertyRequirement'. The form fields include:

- Name:** Property requirement Supplier ID
- Assessment:** checked
- Tested class:** VCARD Organization
- Cleansing:** unchecked
- Tested property:** FOO Supplier ID
- Validation:** unchecked
- Valid from:** 30/05/2012 00:00
- Filtering:** unchecked
- Valid until:** 31/12/2012 00:00
- Importance:** 10
- Unit of importance:** [empty]
- Confidence:** 100
- Unit of confidence:** [empty]
- Task dependent:** unchecked
- Applies for task:** [empty]
- Requirement description:** Every record in VCARD Organization must possess a unique supplier ID starting from 1.
- Requirement source:** Business policy SOP 30/6

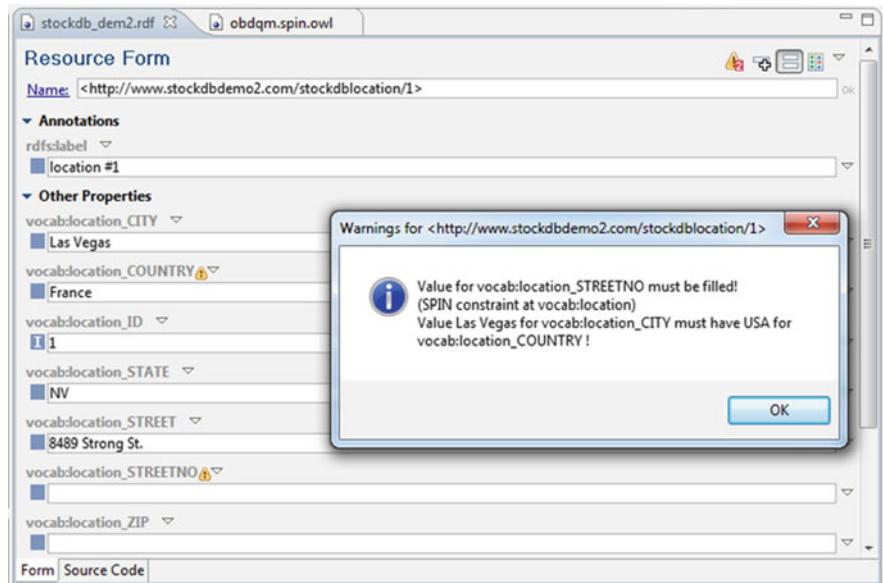
Below these, there are sections for 'Completeness / Uniqueness' (with 'Property required' checked), 'Syntax Rule' (with 'Values must be unique' checked), and 'RequiredValue' checked. There are also sections for 'Legal Value Range' (with 'Lowest legal value' set to 1 and 'Highest legal value' set to 1), 'Legal Value Rule' (with 'Class with legal values' and 'Property with legal values' dropdowns), and 'Illegal Value Rule' (with 'Class with illegal values' and 'Property with illegal values' dropdowns). A summary section at the bottom contains the text 'Summary: CUSTOM REQUIREMENTS'.

**Fig. 5** Form that captures data requirements related to properties and data requirement metadata

be used to evaluate the completeness dimension as part of data quality assessment, i.e., the “process of assigning numerical or categorical values to IQ dimensions in a given setting” [28]. Based on such an assessment report that can be generated via SPARQL queries, people can quickly gain insight into the degree to which the data meets their specified requirements.

Since the major source for data requirements are usually domain experts that do not necessarily know how to program, it is not suitable to express data requirements in programming code such as RDF. Hence, forms can be used to guide the users to express quality-relevant knowledge in a structured format. Figure 5 shows a form that can be used to collect and generate data requirements in the notation provided by the DQM-Vocabulary with a semantic wiki.

Besides data quality monitoring and assessment, the same data requirements can also be used to verify data during data entry to avoid the creation of poor data at its root. When representing data requirements with the SPIN framework as implemented in the TopBraid Composer Software, the requirements can immediately be



**Fig. 6** Example of data validations based on SPIN

used to verify entered information as shown in Fig. 6 [23]. In case of requirement violations, the fields of the entering form containing incorrect data will be flagged and a warning message explaining the reason for the requirement violation will be displayed.

Hence, due to its representation in RDF with the help of standardized vocabularies, the same requirements can be used for multiple data quality management tasks at the same time, such as data quality monitoring, data quality assessment, and data validation. This effective use of the captured data requirements and the possibility to share data requirements on a webscale enable the minimization of manual effort for data quality management, especially in large environments, while helping to support the constructive discussion about different quality perceptions. Moreover, due to its materialization, data requirements can be compared and inconsistencies can easily be identified, e.g., with help of SPARQL queries, as explained in the next section.

### 3.2 Automated Identification of Data Requirement Conflicts

Data requirements can usually be specified by several different parties, such as data consumers, business managers, IT staff, or even by legislative authorities. Hence, there is usually no central authority that defines data requirements. Therefore, data requirements may be contradictory to each other. Due to the importance of data quality management, it is sometimes necessary to generate a harmonized

```

PREFIX dqm:<http://purl.org/dqm-vocabulary/v1.1/dqm#>
SELECT (?s AS ?uniquevaluerq) (?s2 AS ?nonuniquenessreq)
WHERE{
?s a dqm:UniqueValueRule .
?dqmtestedClass ?class1 .
?class1 dqm:hasURI ?class1URI .
?dqmtestedProperty1 ?prop1 .
?prop1 dqm:hasURI ?prop1URI .
OPTIONAL{
?s2 a dqm:PropertyRequirement .
?s2 dqmtestedClass ?class2 .
?class2 dqm:hasURI ?class2URI .
?s2 dqm:testedProperty1 ?prop2 .
?prop2 dqm:hasURI ?prop2URI .
FILTER(str(?prop1URI) = str(?prop2URI) && str(?class1URI) = str(?class2URI) && ?s != ?s2)
MINUS{
?s2 a dqm:UniqueValueRule
}
FILTER(bound(?prop2URI))
}
}

```

**Fig. 7** SPARQL query for the identification of inconsistent unique value requirements

quality perspective on data, especially in closed settings within an organization. Current conventional data quality management tools lack support for the automated identification of data requirements, since they usually represent data requirements as part of programming code. When representing data requirements in RDF, e.g., with help of the DQM-Vocabulary, the requirements can be compared to each other with standard SPARQL queries. Figure 7 illustrates a query that can be used to identify inconsistent unique value requirements.

It must be stressed that there may also be duplicate consistent requirements, which can also be identified via other queries. Hence, data quality management based on materialized data requirements in RDF facilitates the management of consistency between data requirements in very large environments. This is a major distinction from traditional data quality management tools that hide data requirements in programming code.

### 3.3 Semantic Definition of Data

Since schema elements of ontologies are accessible via URLs, additional information about classes and properties may be easily looked up via dereferencing the URL. Most ontology elements that are currently available on the Semantic Web contain a textual definition of the semantics of their classes and properties [29]. Therefore, data providers of the Semantic Web can easily identify elements of other ontologies that can be reused to publish their data. Figure 8 shows a rendered definition of the class “DataRequirement” as retrievable from <http://semwebquality.org/dqm-vocabulary/v1/dqm#DataRequirement>. Opposite to the Semantic Web, many traditional information systems and databases do not directly provide definitions for their schema elements. Moreover, due to the tight relationship between schema and instance data, database schemata are usually not reused, although there are many organizations that have to store almost the same information [30].

**dqm:DataRequirement** (rdf:type owl:Class)

**URI** <http://purl.org/dqm-vocabulary/v1/dqm#DataRequirement>

**rdfs:label** Data Requirement

**rdfs:comment**

A data requirement is a prescribed directive or consensual agreement that defines the content and/or structure that constitute high quality data instances and values.

**is rdfs:domain of** [dqm:appliesFor](#) [dqm:assessment](#) [dqm:cleansing](#) [dqm:confidence](#) [dqm:filtering](#) [dqm:hasScore](#) [dqm:importance](#) [dqm:lastModified](#) [dqm:reqDescription](#) [dqm:reqName](#) [dqm:reqSource](#) [dqm:ruleViolation](#) [dqm:taskDependent](#) [dqm:testedClass](#) [dqm:testedProperty1](#) [dqm:unitOfConfidence](#) [dqm:unitOfImportance](#) [dqm:validFrom](#) [dqm:validUntil](#) [dqm:validation](#)

**is rdfs:range of** [dqm:basedOn](#) [dqm:hasRequirement](#) [dqm:ruleOfIdentification](#)

**Subclasses**

- [dqm:ClassRequirement](#)
- [dqm:CustomRequirement](#)
- [dqm:MultiPropertyRequirement](#)
- [dqm:PropertyRequirement](#)

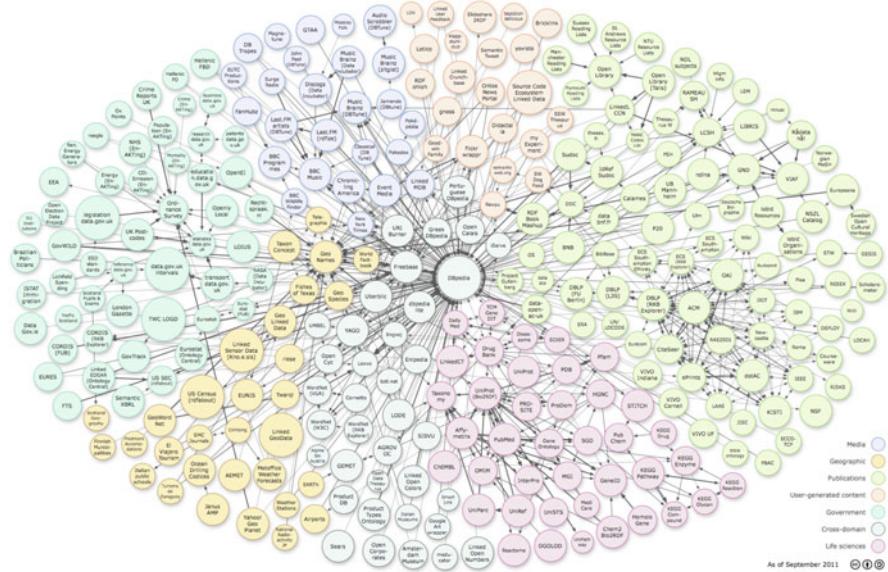
**Fig. 8** Definition of the class “DataRequirement” as provided by the DQM-Vocabulary

It must be stressed that besides this unstructured definition of ontology elements, it is also possible to define formal axioms for the elements of an ontology, e.g., to constrain the members of a class. However, the precision of the semantics and the complexity rises with the degree of formality. Therefore, the degree of formality should be kept at a minimum to facilitate consensual agreement and reuse of ontologies in large environments such as the Web.

### 3.4 Using Semantic Web Data as a Trusted Reference

Besides the technological frameworks, the Semantic Web also provides a huge amount of data for several domains, such as geography, media, and life sciences, which can be used as trusted reference data in data quality monitoring and assessment queries. For example, we can use the city/country combinations of GeoNames, a public knowledge base for geographical data,<sup>11</sup> for monitoring and

<sup>11</sup><http://geonames.org>.



**Fig. 9** Linking Open Data diagram. Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch. <http://lod-cloud.net/> (Retrieved on April 2, 2012)

assessing the quality of address data. Moreover, e-commerce companies have started to publish their product catalogs in RDF format on the Web with the widely known GoodRelations vocabulary, an ontology for e-commerce.<sup>12</sup> Hence, product data could be verified against public product catalogs available on the Web (Fig. 9).

In particular, legal value and functional dependency requirements are data requirements that are especially applicable for using reference data as shown in [31]. If all data is converted to RDF (on the fly or persistently), SPARQL queries can process both data and match the tested data against the trusted reference data. The identified mismatches can be regarded as requirement violations that need further analysis. The use of reference data from the Semantic Web saves a lot of work, since the data already exists and does not have to be created from scratch. However, it must be stressed that there is a high risk to transfer quality errors of the trusted data to the tested data. Hence, it is important to check the reference data for quality problems on a regular basis before using it as a trusted reference. Moreover, Semantic Web data is very heterogeneous and, therefore, may require additional standardization effort before its usage for data quality management tasks.

<sup>12</sup><http://purl.org/goodrelations/>

### 3.5 Content Integration with Ontologies

In today's information system landscapes the integration of information plays a crucial role. Mergers and acquisitions and cooperation with suppliers often drive companies to integrate the data from different systems rather than employing a single system throughout the company or across the whole supply chain. Moreover, consumer data from social media such as Facebook, Twitter, or blogs are used for business analyses, e.g., to evaluate the reputation of certain products. Semantic Web technologies are designed to integrate and link information at Web scale. Therefore, they provide excellent capabilities for content integration. Due to the independence of ontologies from the instance data, it is possible to define conceptual models in a flexible and extendable way. With mapping and conversion tools such as D2RQ<sup>13</sup> or Virtuoso RDF-Views,<sup>14</sup> it is then possible to map the concepts of an ontology to the local schemata of relational databases.

Furthermore, it is also possible to convert other source formats like CSV or XML files to RDF with tools like TopBraid Composer.<sup>15</sup> The provenance of the data can thereby be preserved with help of URI design or annotation of provenance information to the data, e.g., with the provenance vocabulary.<sup>16</sup> The definition of hierarchies, i.e., subclass relationships, in ontologies facilitates content analysis at several different levels satisfying multiple perspectives. Moreover, it is possible to explicitly model equivalence relationships. This is especially useful to analyze a nearly complete corpus of data without knowing all synonym relationships. Additionally, unstructured information such as documents or regular Web sites can be easily linked via ontologies to the structured data and, therefore, create a fully integrated information system. The linkage of relevant unstructured information may thereby be supported by natural language processing technology that automatically categorizes unstructured data, such as OpenCalais.<sup>17</sup> Figure 10 sketches an architecture in which ontologies are used as a global schema to integrate data from heterogeneous database schemata, XLS files, documents, the conventional Web, and the Semantic Web.

The integration capabilities of ontologies are very promising as indicated by several research approaches in this area [7, 32–38]. However, the process of modeling a precise semantic network is a complex analytical task. Additionally, the ontology as a global schema for information integration requires the creation of a consensual agreement among its users. With increasing precision and formality of ontologies, this consensual agreement may often be difficult to accomplish. Moreover, it is reported that users have difficulties when familiar concepts from

---

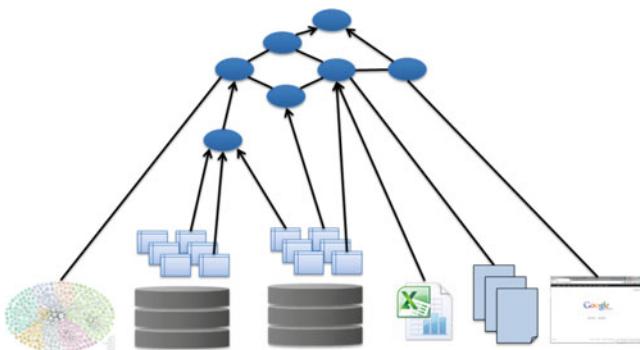
<sup>13</sup><http://d2rq.org/>.

<sup>14</sup><http://virtuoso.openlinksw.com/whitepapers/relational%20rdf%20views%20mapping.html>.

<sup>15</sup>[http://www.topquadrant.com/products/TB\\_Composer.html](http://www.topquadrant.com/products/TB_Composer.html).

<sup>16</sup><http://purl.org/net/provenance/ns>.

<sup>17</sup><http://www.opencalais.com>.



**Fig. 10** Content integration architecture with a global ontology

local database views disappear behind ontologies (cf. [6]). However, the integration capabilities of ontologies and the Semantic Web may help to improve master data management architectures, since the semantics of the local schemata can be preserved and put into explicit relationships among each other. First attempts have been made by IBM research towards this direction as published in [39]. Moreover, business intelligence architectures can benefit from the integration capabilities of Semantic Web architectures and improve the completeness of information by integrating information from the WWW, documents, or the Semantic Web.

## 4 Limitations of Semantic Web Technologies for DQM

As outlined in the previous sections, the Semantic Web provides promising technologies to improve the support data quality management. However, the employment of Semantic Web technologies for data quality management contains several pitfalls that can be avoided with a clear concept and understanding.

Ontologies are very expressive and can, therefore, be modeled with highly formal axioms leading to a complexity that makes the creation of a shared understanding impossible. If the degree of formality stays in the limits of the necessary and the textual definitions of ontological concepts are maintained, then the complexity of ontologies stays controllable. To increase the speed of adoption elements of ontologies should be mapped to familiar conceptual elements, if possible.

Moreover, the conversion of relational data into RDF triples increases the amount of data due to its structure. Scalable computing technologies such as in-memory computing or MapReduce algorithms should be able to cope with this challenge. Moreover, regular performance benchmarks of triplestores show significant performance increases (cf. [40]).

Finally, techniques for annotating unstructured information often do not go beyond the assignment of tags. In many business cases, the relevant domain

vocabularies are still missing. Therefore, the integration of unstructured information still requires some maturity to serve in business information integration. However, the increasing availability of commercial products and the increasing support of Semantic Web data, e.g., recently by IBM and Oracle, show that Semantic Web technologies have reached a level of maturity that should be sufficient for industrial projects not only in the area of data quality management.

## 5 Summary and Future Directions

In this chapter, we have outlined several different ways how Semantic Web technologies may improve data quality management. In particular, we have shown how data requirements can be represented in RDF with help of a standardized vocabulary and thereby facilitate automated (1) identification of requirement violations, (2) generation of data quality scores, (3) validation of data entries, and (4) identification of contradictory requirements. Besides this exploitation of data requirements for data quality management, the materialized data requirements can be shared on the Web, e.g., across multiple different organizations with a supply chain. This helps to create a consensual agreement about the different quality perspectives on data. Moreover, we have shown that semantic definitions of conceptual elements of ontologies can easily be published, which may increase their reuse and reduce misinterpretations of data.

Additionally, we have sketched the broad integration capabilities of ontologies in distributed environments which allows the integration of data and documents. Hence, integrating data and its documentation may support integrated data quality management and business process management, since data requirements can be combined with standard operating procedures and other process documentation. Moreover, the integration capabilities of Semantic Web architectures allow the integration of data from different sources with preservation of their original semantics and provenance. This may help to generate a higher degree of trust in the data. Furthermore, the Semantic Web itself already provides a lot of data that can be used as reference data, e.g., to evaluate the quality of location-based data.

In summary, there are multiple areas in which Semantic Web technologies and the Semantic Web itself provide promising features regarding the management of data quality. However, little experience has been collected in the practical application of Semantic Web artifacts. Moreover, the presented scenarios may not fully exploit the potential use cases of Semantic Web technologies for data quality management. Hence, future work should address the application of Semantic Web technologies in the presented areas to real-world settings, while remaining open for further use cases in which Semantic Web technologies may be applied for improving data quality. Moreover, the quality of data on the Semantic Web itself remains an open topic that needs to be addressed by research, especially when attempting to use Semantic Web data as trusted reference for data quality management.

## References

1. Berners-Lee T, Hendler J, Lassila O (2001) The Semantic Web. *Scientific Am* 284(5):34–43
2. Beckett D (2004) RDF/XML Syntax Specification (Revised). World Wide Web Consortium (W3C). <http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/>. Accessed 14 Aug 2010
3. Gruber TR (1993) A translation approach to portable ontology specifications. *Knowl Acquis* 5(2):199–220
4. Uschold M, Gruninger M (1996) Ontologies: principles, methods, and applications. *Knowl Eng Rev* 11(2):93–155
5. Sahoo SS, Halb W, Hellmann S, Idehen K, Thibodeau T, Auer S, Sequeda J, Ezzat A (2009) A survey of current approaches for mapping of relational databases to RDF. W3C RDB2RDF Incubator Group. [http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF\\_SurveyReport\\_01082009.pdf](http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport_01082009.pdf). Accessed 4 Jan 2012
6. Leser U, Naumann F (2007) *Informationsintegration: Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen*. 1st edn. dpunkt-Verlag, Heidelberg
7. Alexiev V, Breu M, de Bruin J, Fensel D, Lara R, Lausen H (2005) Information integration with ontologies: experiences from an industrial showcase. Wiley, Chichester
8. Wang RY, Strong DM (1996) Beyond accuracy: what data quality means to data consumers. *J Manage Inf Syst* 12(4):5–33
9. Redman TC (2001) Data quality: the field guide. Digital Press, Boston
10. Kahn BK, Strong DM, Wang RY (2002) Information quality benchmarks: product and service performance. *Commun ACM* 45(4):184–192. doi:<http://doi.acm.org/10.1145/505248.506007>
11. Olson J (2003) Data quality: the accuracy dimension. Morgan Kaufmann; Elsevier Science, San Francisco
12. ISO (2009) ISO 8000–102:2009. Data quality - Part 102: Master data: Exchange of characteristic data: Vocabulary. ISO, Switzerland
13. English LP (1999) Improving data warehouse and business information quality: methods for reducing costs and increasing profits. Wiley, New York
14. Wang RY (1998) A product perspective on total data quality management. *Commun ACM* 41 (2):58–65. doi:<http://doi.acm.org/10.1145/269012.269022>
15. Wand Y, Wang RY (1996) Anchoring data quality dimensions in ontological foundations. *Commun ACM* 39 (11):86–95. doi:<http://doi.acm.org/10.1145/240455.240479>
16. Rahm E, Do H-H (2000) Data cleaning: problems and current approaches. *IEEE Data Eng Bull* 23(4):3–13
17. Oliveira P, Rodrigues F, Henriques PR (2005) A formal definition of data quality problems. Paper presented at the International Conference on Information Quality (MIT IQ Conference) Cambridge, MA, November 10–12
18. Oliveira P, Rodrigues F, Henriques PR, Galhardas H (2005) A taxonomy of data quality problems. Paper presented at the 2nd Int. Workshop on Data and Information Quality (in conjunction with CAiSE'05), Porto, June 14, 2005
19. ISO (2005) ISO 9000:2005. Quality management systems - fundamentals and vocabulary, vol TC 176/SC. International Organization for Standardization,
20. Codd EF (1982) Relational database: a practical foundation for productivity. *Commun ACM* 25(2):109–117. doi:<http://doi.acm.org/10.1145/358396.358400>
21. Codd EF (1990) The relational model for database management: version 2. Addison-Wesley, Reading
22. Bizer C, Cyganiak R (2009) Quality-driven information filtering using the WIQA policy framework. *Web Semantics* 7 (1):1–10. doi:<http://dx.doi.org/10.1016/j.websem.2008.02.005>
23. Fürber C, Hepp M (2010) Using SPARQL and SPIN for data quality management on the Semantic Web. Paper presented at the 13th International Conference on Business Information Systems 2010 (BIS2010), Berlin, May 3–5

24. Fürber C, Hepp M (2011) SWIQA – A Semantic Web information quality assessment framework. Paper presented at the European Conference on Information Systems (ECIS) 2011, Helsinki, 9–11 June 2011
25. Brüggemann S, Aden T (2007) Ontology based data validation and cleaning: restructuring operations for ontology maintenance. 37. Jahrestagung der Gesellschaft für Informatik e.V., vol 109. GI,
26. Fürber C, Hepp M (2011) Towards a vocabulary for data quality management in Semantic Web architectures. Paper presented at the 1st International Workshop on Linked Web Data Management, Uppsala, 21–25 March 2011
27. Antoniou G, Van Harmelen F (2008) A Semantic Web primer. Cooperative information systems, 2nd edn. MIT Press, Cambridge
28. Ge M, Helfert M (2007) A review of information quality research - develop a research agenda. Paper presented at the 12th International Conference on Information Quality (ICIQ), USA, November 9–11
29. Hogan A, Harth A, Passant A, Decker S, Polleres A (2010) Weaving the pedantic Web. Paper presented at the WWW2010 Workshop on Linked Data on the Web (LDOW 2010), Raleigh, 27 April 2010
30. Berners-Lee T (1998) Relational databases on the Semantic Web. <http://www.w3.org/DesignIssues/RDB-RDF.html>. Accessed 5 Jan 2012
31. Fürber C, Hepp M (2010) Using Semantic Web resources for data quality management. Paper presented at the 17th International Conference on Knowledge Engineering and Knowledge Management (EKAW2010), Lisbon, October 11–15, 2010
32. Fensel D (2002) Intelligent information integration in B2B electronic commerce. Kluwer international series in engineering and computer science, vol 710. Kluwer Academic, Boston
33. Kokar MM, Matheus CJ, Baclawski K, Letkowski JA, Hinman M, Salerno J (2004) Use cases for ontologies in information fusion. Paper presented at the 7th International Conference on Information Fusion, Stockholm, 28 June to 1 July 2004
34. Niemi T, Toivonen S, Niinimaki M, Nummenmaa J (2007) Ontologies with Semantic Web/Grid in Data Integration for OLAP. *Int J Semant Web Inf Syst* 3(4):25–49
35. Perez-Rey D, Anguita A, Crespo J (2006) OntoDataClean: ontology-based integration and preprocessing of distributed data In: Biological and medical data analysis, vol 4345/2006. Lecture Notes in Computer Science. Springer, Berlin, pp 262–272
36. Skoutas D, Simitsis A (2007) Ontology-based conceptual design of ETL processes for both structured and semi-structured data. *Int J Semant Web Inf Syst* 3(4):1–24
37. Souza D, Belian R, Salgado AC, Tedesco PA (2008) Towards a context ontology to enhance data integration processes. ODBIS
38. Wache H, Voegele T, Visser U, Stuckenschmidt H, Schuster G, Neumann H, Huebner S (2001) Ontology-based integration of information - a survey of existing approaches. Paper presented at the IJCAI-01 Workshop on Ontologies and Information Sharing, Seattle, 4–5 August 2001
39. Wang X, Sun X, Cao F, Ma L, Kanellos N, Zhang K, Pan Y, Yu Y (2009) SMDM: enhancing enterprise-wide master data management using Semantic Web technologies. *Proc VLDB Endow* 2(2):1594–1597
40. Bizer C, Schultz A (2011) Berlin SPARQL benchmark (BSBM) results (February 2011). <http://www4.wiwiiss.fu-berlin.de/bizer/BerlinSPARQLBenchmark/results/V6/index.html>. Accessed 4 Jan 2012

# Data Glitches: Monsters in Your Data

Tamraparni Dasu

**Abstract** Data types and data structures are becoming increasingly complex as they keep pace with evolving technologies and applications. The result is an increase in the number and complexity of data quality problems. Data glitches, a common name for data quality problems, can be simple and stand alone, or highly complex with spatial and temporal correlations.

In this chapter, we provide an overview of a comprehensive and measurable data quality process. To begin, we define and classify complex glitch types, and describe detection and cleaning techniques. We present metrics for assessing data quality and for choosing cleaning strategies subject to a variety of considerations. The process culminates in a “clean” data set that is acceptable to the end user. We conclude with an overview of significant literature in this area, and a discussion of opportunities for practice, application, and further research.

## 1 Introduction

In the era of iPad and Facebook, where both information and entertainment are increasingly electronic, the types and volumes of data being generated are mind-boggling. Scientific and industrial applications, data networks, financial transactions, and mobility applications—the common thread that runs through all these domains of application is the quality of data. With increasing reliance on automation, the types of issues that plague data have become more frequent and complex. They range from the simple and correctible, to problems that might not be identifiable, let alone remediated. Data quality issues can seriously skew the results of data mining and analysis, with consequences that can potentially cost

---

T. Dasu (✉)

AT&T Labs Research, 180 Park Avenue, Florham Park, NJ 07932, USA

e-mail: [tamr@research.att.com](mailto:tamr@research.att.com)

billions; corporations could make erroneous decisions based on misleading results, and machinery could be incorrectly calibrated leading to disastrous failures.

Data issues, if left unaddressed, get entrenched. They propagate rapidly across data systems, contaminating data in an exponential manner. The window of opportunity to clean data is often quite small because the original data might be unavailable later to reconstruct a clean version of the data. For example, if there are errors during the transmission of data, there is only a very short window of time, typically 24 hours, to request retransmission before access to the original data is lost forever. Therefore, it is important to detect and treat data quality issues in a timely fashion.

We introduce below a statistical approach to data quality assessment and cleaning. Given that data quality is highly application specific and domain dependent, we describe a flexible framework within which metrics and solutions can be customized appropriately.

## 1.1 A Statistical Notion of Data Quality

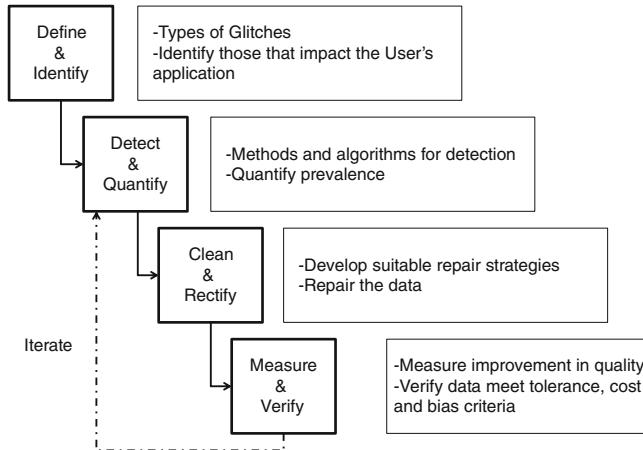
Let  $D = \{d_{ij}\}$  be the data matrix that is available to the user, where  $i = 1, \dots, N$  represents a row that corresponds to a unique entity, and column  $j = 1, \dots, d$  corresponds to an attribute. In reality,  $D$  can take more complex forms, but the following discussion applies irrespective of the exact structure of  $D$ . In our specification,  $D$  has  $N$  records or *samples* and  $d$  attributes or *dimensions*.

In general, the data  $D$  are generated by a real-world process  $P$  such as a data network or financial transaction stream. The user has a certain perception of the data,  $D^*$ , derived from either experience or domain knowledge of  $P$ . The perception reflects the user's beliefs, assumptions, and understanding of the data. For example, a user might expect that a particular column in the data matrix  $D$  corresponds to *revenues measured in dollars and that it follows a log-normal distribution*. Any and all of these could be wrong due to flawed documentation or incorrect model assumptions. The attribute, in reality, could represent *production costs, measured in Euros, transformed to follow a Normal distribution*.

The disparity between the actual data  $D$  and the user's perception of the data  $D^*$  is a measure of the quality of data. The farther  $D$  and  $D^*$  are, the poorer the quality of data, less reliable the results, and less usable the data. The closer  $D$  and  $D^*$  are, the better the quality of data and more reliable the results. The goal of data cleaning or data repair is to bridge the disparity between  $D$  and  $D^*$ .

However, data cleaning or repair should not be performed indiscriminately. The process of cleaning should preserve the original statistical distributional properties and not distort or transform them to such an extent the data are no longer representative of the real-world process  $P$ . Inference based on excessively transformed data could be meaningless or, worse, misleading.

In an ideal world, we should be able to clean the data and bring  $D$  and  $D^*$  closer, without distorting it too much. In the real world, it is more often a trade-off



**Fig. 1** Overview of data quality process: the process is flexible to allow the user to customize data quality assessment and repair to suit resource and accuracy requirements

between cleaning and distortion, within the user’s resource limitations and quality specifications. In this chapter, we provide a brief introduction to these concepts and explain data quality assessment within the framework of statistical theory.

The rest of the chapter is organized as follows. Section 2 provides an overview of the statistical data quality process, emphasizing its iterative nature. In Sect. 3, we discuss new types of glitches and complex glitch patterns that can be leveraged to develop efficient context-dependent cleaning strategies. In Sect. 4, we describe methods for detecting different types of glitches, while Sect. 5 focuses on assessing the quality of data. Section 6 offers suggestions for cleaning and repairing data. In Sect. 7, we discuss how to choose a strategy from a multitude of potential strategies, based on the notion of the trade-off between cleaning and distortion. A brief literature overview is provided in Sect. 8. Finally, we present our conclusions in Sect. 9.

## 2 Data Cleaning, an Iterative Process

Data cleaning is an iterative process. Typically, a user is presented with a data set  $D$  and would like to clean it to meet cleanliness specifications (“no more than 10 % missing values”) subject to resource and cost considerations. The process of cleaning could potentially introduce new glitches and make the data dirtier. Therefore, after each cleaning step, the user remeasures the quality and repeats the process until the specifications are satisfied.

A schematic version of the data quality process is shown in Fig. 1. It has four broad stages. We give a brief overview below and describe each stage in detail in the following sections.

**Define and Identify:** The first step in the data quality process is to have a well-defined notion of what constitutes a data glitch. While there are established definitions for types of glitches (missing, duplicates, inconsistent), it is important to identify those that are relevant to the particular user and context. One user's glitch could be another user's treasured measurement. For example, an extreme observation, known as an outlier or anomaly, could skew one particular user's analysis, because she is interested in computing averages. The outlier would constitute a glitch and should be treated. On the other hand, another user might specifically want those extreme values. He might be interested in rare but high-risk events, like unusual loads on a data network, and would want to preserve them to better understand their occurrence patterns and their impact on network performance.

**Detect and Quantify:** Once the set of relevant glitches is defined, we need methods that can comb through the data and detect the glitches and tag them. *Glitch detector functions* are used to identify glitches and associate with each data cell  $d_{ij}$ , a vector of indicators that denote the presence or absence of each type of glitch. The resulting bit vector  $v_{ij}$  is called a *glitch signature*. Examples of glitch detector functions range from the simple, like the binary presence or absence of an attribute, to the complex, such as a multidimensional outlier detection technique. Glitch signatures can be complex as well and need not be limited to bit vectors. For instance, in the case of outlier detection, they could potentially include  $p$ -values instead of a binary indicator of presence or absence of an outlier.

Once all the glitches are identified and tagged, we can quantify their prevalence and measure the quality of the data using an objective measurement called a *glitch score* and *glitch index*, a weighted average of the glitch signatures. Both glitch signatures and glitch scores can be customized to the needs of the user.

**Clean and Rectify:** It is not essential that the user should clean all the glitches. Some glitches are more expensive to clean than others, and some have little or no impact on the user's applications. Since for each type of glitch there are numerous options for cleaning, the user selects a suitable cleaning method. In the absence of prior knowledge, an experimental framework can be used to choose an appropriate strategy from a set of candidate strategies. The methods are then applied to repair or clean the data.

**Measure and Verify:** The glitch score is recomputed after cleaning, to see whether it meets the user's specified tolerance limits in terms of cleanliness. In addition, we also quantify the impact of data cleaning on the statistical distribution of  $D$  using the notion of *statistical distortion* which is measured by means of a histogram distance. If both the glitch score and statistical distortion are satisfactory, the cleaning process is over and the data are ready to use. Otherwise, the entire process is repeated, subject to user's cost considerations.

We will elaborate on each of the stages of the data quality process in the sections that follow.

### 3 Complex Data Glitches

There is a vast amount of literature on types of data glitches. We provide a brief summary of select references in Sect. 8. In general, data glitches are errors in the measurement and recording of data that adversely impact analysis. We begin our discussion with some examples of common data quality problems.

**Missing Data:** Missing data are instances where individual data values, entire records, and entire attributes disappear, due to a multitude of reasons: human error; malfunctioning software used for data collection, processing, and storing; restrictive data entry interfaces; and system errors. Given the volume of data generated in most modern applications, missing data could go unnoticed for long periods of time. If not addressed, missing data could result in serious biases in the analyses.

**Inconsistent and Faulty Data:** Inconsistencies arise in the data in many ways. Data are entered incorrectly, often due to manual error, and could be hard to detect. For example, when “09-07-2010” is recorded as “09-07-2001,” it is very difficult to tell the entry apart from other legitimate entries, unless we have additional information that flags it as an error. Without a scalable method for validation, there is no way to determine whether the value is correct or not. Given the volume of data, manual checking is not a scalable method.

A variation of this error that is more easily detected is the case where 09/07/2010 is entered as 90/07/2010. We can use the constraint “If month = 07, then the day of the month must lie between 1 and 31” to identify this inconsistency. Constraint satisfaction methods are used widely to automate well-defined consistency checks by ETL (Extract–Transform–Load) tools.

**Anomalies and Outliers:** Data points that are unexpected, while not necessarily incorrect or inconsistent, are potentially suspicious. For example, in the sequence of numbers “1, 2, 8, 3, 0, 2, 3, 4, 1, 6, 123, 1, . . . ,” 123 stands out as an unexpected value. Considerable literature in statistics and computer science has been dedicated to the detection and removal of unexpected data, for example [1, 4].

**Duplicates:** Data records that claim to represent the same piece of information but have distinct, nonunique values are considered to be duplicates. For example, in a database where name and address uniquely identify an individual and the attribute of interest is height, the following two nonunique records are duplicates.

John Doe, 123 Main Street, Tall  
John Doe, 123 Main Street, Short

Duplicates are often hard to define, because the notion of uniqueness of an entity varies from context to context within the same framework.

**Undocumented Data:** One of the most egregious data quality problems is the lack of documentation. Without a proper data dictionary, what do the values in the following data record mean?

If we know that the data pertain to a telephone record, we can guess that the first field corresponds to a telephone number, the second to a dollar amount, and the third to utilization in hours. But it would be a guess, and without documentation, the data are *unusable*.

There are numerous other data quality issues, including a variation in quality across different sections of the data when data are integrated from sources with different recency. Some sections of the data could contain outdated information. Data integration itself could introduce data quality issues by joining together records that belong to different entities even though they have the same supposedly “unique” join key.

### **3.1 A Challenging Problem**

Detecting and treating data quality problems is challenging due to complexity and application-specific nature; it is difficult to generalize or automate solutions. Other reasons include:

**Relevance:** The meaning, severity, and treatment of data glitches vary from domain to domain. A marketer might be interested in typical values, while an engineer might be interested in outliers. The latter will have a lower tolerance to missing values while the former might not care as long as there is a reasonable sample to estimate typical values. This lack of generality makes it hard to automate data quality solutions across domains.

**Ambiguity:** Sometimes, the boundary between good and bad data is not very clear. Different outlier detection methods could specify different thresholds for identifying abnormal values, resulting in potentially conflicting outcomes.

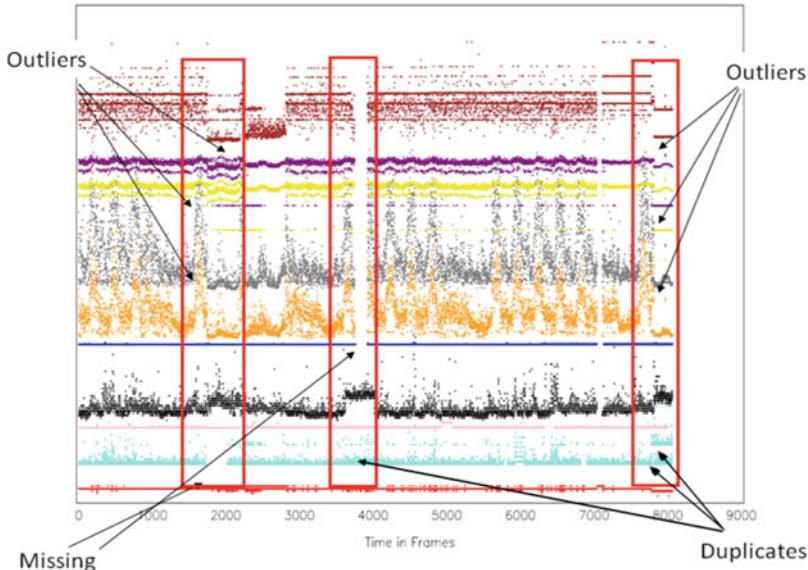
**Complex dependence:** Different types of data glitches co-occur, or one type of glitch could mask the occurrence of others. For example, missing values could mask the presence of duplicates or outlying values.

**Dynamic:** In addition, as data evolve, the extent and type of glitches change over time, making glitch detection models obsolete.

In recent work [3], Berti-Equille et al. propose that glitches tend to be complex entities that co-occur in distinct patterns, as opposed to the common assumption of independent and random occurrence of glitches. As an example, consider Fig. 2. It shows ten attributes (differentiated by shading) related to the performance of a network element in an IP network over a brief period. The attributes are plotted on a transformed scale. Missing values, outliers, and duplicates co-occur, sometimes with a short lag in time.

To capture the variety and multiplicity of these interdependent co-occurring glitches, the authors propose the following classification of complex types of glitches.

Given a data set  $D = \{d_{ij}\}$ :



**Fig. 2** Complex glitches: in a data stream of IP network data collected over a brief period, multiple glitches occur in complex patterns of static and temporal dependence. The plot shows ten attributes (differentiated by shading) on a transformed scale. Glitches are of multiple types, occurring at multiple times, affecting multiple attributes and multiple records

**Multi-type Glitch:** A value  $d_{ij} \in D$  has a multi-type glitch if there are at least two categories of glitches associated with it.

**Concomitant Glitches:** Two or more values  $d_{ij}$  and  $d_{ij'}$  ( $j \neq j'$ ) in  $D$  are concomitant glitches if they are both glitches and occur in the same data row  $i$ .

**Multi-occurred Glitch:** A multi-occurred glitch  $c$  in the data set  $D$  is a glitch whose type is shared by two or more distinct values in the data set.

The authors of [3] demonstrate that patterns of dependence and co-occurrence of glitches can be leveraged to develop highly efficient, quantitative cleaning strategies that are specially tailored for the data set. By cleaning co-occurring glitches simultaneously rather than each glitch type independently, the user saves on resources and cycle times.

## 4 Glitch Detection

Glitch detection, like most aspects of data quality, is dependent on the application and the user. There are many tools in the fields of data mining, database theory, and statistics to identify a wide variety of problems in the data. It could be as simple as identifying missing values (“value is not populated”) or as complex as identifying

inconsistent values by checking against an exhaustive set of rules specified by domain experts.

In general, glitch detection methods fall into two broad categories: constraint-based methods developed in consultation with experts or constructed from data properties and functional dependencies that the data must satisfy, and quantitative methods based on statistical and data-mining approaches.

In our experience, an effective data quality auditing methodology must have both components. Data quality is highly domain and context dependent, and therefore it is important to incorporate domain knowledge gathered from experts into a set of rules or constraints that the data must satisfy. In addition, given the vast amounts of rapidly accumulating data, statistical techniques are essential to screen the data and isolate smaller subsets for further analyses to identify patterns and co-occurrences. Such glitch patterns could aid root cause identification to eliminate future glitches and guide effective data repair.

## ***4.1 Constraint Satisfaction Methods***

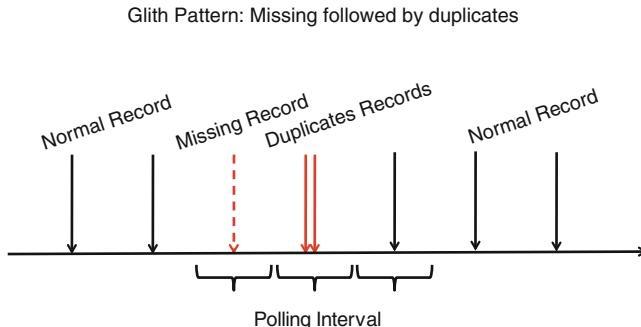
Constraint satisfaction procedures enforce compliance with specific constraints. Some constraints can be formulated based on heuristics or consultation with experts. For example, “duration should be nonnegative,” or “if US ZIP code = 07932, then state = NJ,” or “if wireless service = voice only, then text traffic = 0.” Any record that violates this rule is deemed inconsistent.

As another example, consider the constraint, “unique identifier should be present.” A unique identifier depends on the application and is typically used to denote distinct samples, entities, or measurements in the data, loosely speaking. For example, a credit card number and a time stamp together would constitute a unique identifier for a data stream of credit card transactions. Any record that does not meet this constraint could be considered missing, because the data measurements that are populated (“items bought, amount paid in dollars”) cannot be attributed to a distinct entity (“credit card number and transaction time”).

Unique identifiers are critical for identifying duplicates. Duplicate detection can be posed as a constraint as well. “If two or more records have the same unique identifier, then the two records should be nearly identical in the remaining attributes.” Otherwise, they constitute duplicates that need to be remediated.

## ***4.2 Statistical Methods***

Statistical methods are most commonly employed for detecting inconsistencies and outliers. There is an abundance of outlier detection methods, such as those that employ error bounds, quantile-based methods, and model-based methods. Most are univariate, that is, they detect outliers in each attribute separately. Common



**Fig. 3** Pattern of glitches: missing values (*broken arrow*) are followed by duplicates (*solid arrows close together*). By imputing a missing value using the nearest duplicate (in time), we rectified two types of glitches at once and kept statistical distortion low by using real values rather than statistical estimates like the mean or median

univariate methods include  $3\sigma$  limits of a Gaussian distribution, quantile-based methods such as the 5th and 95th percentiles of a distribution, and outliers with respect to statistical models such as regression-based outliers.

Multivariate methods take into account the interdependence between attributes. A common multivariate outlier detection method uses the Hotellings  $T^2$  statistic, the multivariate Gaussian equivalent of the  $3\sigma$  test. Details of this method can be found in [12].

### 4.3 Patterns of Glitches

In addition to detecting individual glitches, it is important to detect glitch patterns and glitch dependence. Glitch patterns offer clues to efficient data-cleaning strategies that are specific to the application.

Figure 3 depicts a data stream generated by a data network where a data record is scheduled to arrive in every polling interval. However, we noticed that in a large proportion of instances, missing values (dashed arrows) in a polling interval were immediately followed by duplicates (solid arrows) in the next polling interval. By consulting domain experts (network engineers), we found that an occasional delay in polling causes this pattern. Therefore, instead of using a blind cleaning strategy of imputing missing values using a mean or a median, we imputed missing values using the closest duplicate (in time) from the next polling interval. Thus, we were able to (1) treat two types of glitches (missing and duplicates) simultaneously and (2) make good use of valuable data collected at great expense, rather than override it with estimated (mean, median) or synthetic (simulated) values. The paper [3] contains an exhaustive discussion of how to use glitch patterns and dependence structure to develop cleaning strategies.

## 5 Quality Assessment

As in most aspects of data quality, data quality assessment is highly context and application dependent. It is difficult to formulate a general solution that will work in all situations. In this section, we outline a high-level approach that can be customized to the needs of a specific user. There are three main steps: (a) annotating the glitches, (b) weighting them according to the user's prioritization, and (c) combining them into a single comprehensive score that can be used to evaluate quality, and measure the improvement in quality. The user can compare strategies using the glitch score and choose the most suitable one.

### 5.1 Glitch Signatures

A glitch signature is a way of summarizing the usability of a value in the data matrix. Each data element  $d_{ij} \in D$  is mapped to a vector of bits  $v_{ijk}, k = 1, \dots, c$ , where each bit corresponds to one of the  $c$  glitch types. For example, suppose that we are interested in  $c = 3$  types of glitches—missing values, outliers, and inconsistencies, in that order. In addition, suppose that the value  $d_{ij}$  has an inconsistency associated with it. The corresponding vector of bits that indicates the presence or absence of each of these glitch types in  $d_{ij}$  is given by

$$v_{ij} = (0, 0, 1). \quad (1)$$

The bit vector  $v_{ij}$  is called the *glitch signature* of  $d_{ij}$ . While this is a very simple example, see [3] for a more nuanced formulation of glitch signatures.

A value  $d_{ij} \in D$  is “clean” if and only if every value in  $v_{ij}$  is zero, that is, the sum of the bits is 0. Glitch signatures can be considered *augmented* or derived data that enable us to flag glitches, quantify their prevalence, and identify patterns in an analytic, automated fashion.

### 5.2 Glitch Weighting and Scoring

*Glitch weighting* allows the user to choose the importance of each type of glitch. Let  $\{w_k\}, k = 1, \dots, c$  be values between 0 and 1 such that  $\sum_k w_k = 1$ . If the user wishes to eliminate missing values but is not too concerned about outliers, then the weights can be chosen to reflect that. For instance,  $w_1 = 0.6$ ,  $w_2 = 0.1$ , and  $w_3 = 0.3$  for the three types of glitches of missing, outliers, and inconsistencies reflect their relative importance to the user.

The individual glitch score  $g_{ij}$  associated with each value in the data matrix is computed as follows:

$$g_{ij} = \sum_k v_{ijk} w_k. \quad (2)$$

For the above example, the glitch score would be

$$0 \times 0.6 + 0 \times 0.1 + 1 \times 0.3 = 0.3$$

### 5.3 Glitch Index

The overall glitch score  $G(D)$  of the data set  $D$ , called *Glitch Index*, would simply be

$$G(D) = \sum_{ij} g_{ij}. \quad (3)$$

Note that we can extend the notion of weighting to individual attributes (columns) and rows (records or samples) by devising additional weighting schemes similar to the one above. A network engineer might consider glitches in CPU usage data of a router to be more important than the latitude and longitude information of the physical location of a piece of equipment. The latter does not change often and can be inferred easily from prior values.

## 6 Data Repair

Once the glitches have been tagged and quantified, we are ready to clean the data. Many considerations go into choosing a suitable strategy. The multitude of methods and tools available for this purpose is quite daunting. We give a brief overview below. The ultimate choice depends on the user's requirements.

### 6.1 Cleaning Methods

Glitch detection methods often include a provision for cleaning the data. A comprehensive overview of data-cleaning methods can be found in Chap. 5 of [5]. Additional material can be found in [2]. We describe the salient points below.

**Missing Value Imputation and Treatment of Inconsistencies:** Missing value imputation methods can range from the simple to highly complex, providing a perfect example of the wide spectrum of methods available. The simplest approach consists of replacing missing values in a numeric attribute by a *statistical estimate* such as a mean or a median of the non-missing values of that attribute. Or, we can take into account an attribute's relationship to other attributes by using *regression models* to "predict" the value of the missing attribute as a function of other non-missing attributes. A more exhaustive approach would entail simulating a joint distribution of all the attributes and imputing values drawing from the simulated

distribution. Statistical software SAS includes the module PROC MI for missing value imputation.

Occasionally, a default value, such as 9,999 or  $-10,000$  is used to denote missing values. Usually, these are easy to detect, unless a common value like 0 is used to denote missing values, or there is no standard representation, resulting in multiple ways of denoting missing values. A data browser like Bellman [6] can be used to identify such nonstandard representations.

Non-numeric attributes can be imputed using heuristics and functional dependencies. For example, “if ZIP = 07932 and STATE = missing, then impute STATE = New Jersey.”

There is a vast amount of literature on ETL cleaning tools in the database community. See tutorial [2] for details. In the absence of domain-specific solutions, missing value imputation methods can often be used to correct inconsistencies.

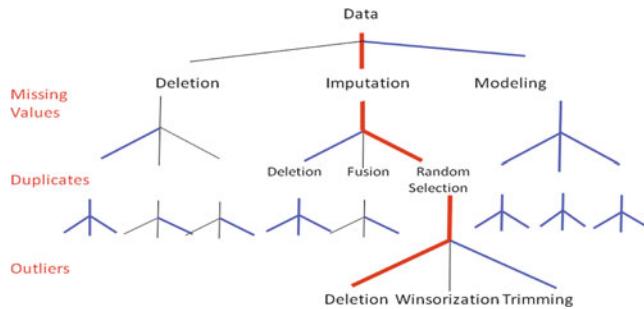
**Outlier Treatment** Outliers are data values that are unexpected, based either on likelihood of occurrence or on other criteria. Whether outliers should be treated, or preserved in the data, depends on the user and application. Because outliers tend to be few in number, the easiest solution is to simply *drop* them from the data set. If that is not an option, a common approach in statistics is *Winsorization*, where outliers are replaced with the nearest non-outlying value of the attribute. For example, if any value beyond the  $3 - \sigma$  is considered an outlier, then every outlier is replaced with the closest  $3 - \sigma$  limit.

Other approaches may entail treating outliers separately while developing models such as regression.

**De-duplication** There is extensive literature in the database community on the treatment of duplicate records. See Elmagarmid et al. [4] for an overview of duplicate detection in databases. Data repair consists of eliminating the duplicates by means such as retaining a random record or fusing the duplicates into a single unique record as discussed in [11].

For example, if there are multiple usage records associated with a given router on a network with the same time stamp, we could (1) retain the first (or last or middle) record, or (2) choose a random record each time, or (3) combine all the records for the router with the same time stamp using averaging techniques.

**Pattern driven methods** In Sect. 4.3, we discussed data-driven strategies that are developed using glitch patterns. Data-mining methods and analytical techniques are used to identify these patterns. Domain experts examine the patterns and participate in developing cleaning strategies. This is an excellent way to customize solutions to specific problems. Even in the absence of domain experts, the dependence between glitches and their correlation patterns still offer valuable clues to glitch treatment as discussed in [3].



**Fig. 4** A simplistic cleaning strategy tree: with a sequential, predefined, “one at a time” approach to treating data glitches, the number of possibilities is exponentially large. In this example, we consider three treatment possibilities for each of three data glitches (missing, duplicates, and outliers) and arrive at 27 possible strategies

## 7 Choosing Data-Cleaning Strategies

We have used the phrases data-cleaning method and data-cleaning strategy interchangeably. However, to be more specific, a strategy is a concatenation of methods. For example, we might start by imputing missing values, then remove duplicates, and finally treat outliers, a strategy that is a concatenation of three methods. Even with as few as three types of glitches, different methods of treatment can lead to an exponentially large number of possible strategies as illustrated in Fig. 4. Given that the order of treatment has an influence on the outcome, the possibilities are even more overwhelming.

In [7], Dasu and Loh suggest using a three-pronged approach to evaluating data-cleaning strategies that takes into account: (1) improvement in glitch scores, (2) impact of the strategy on statistical distribution of the original data, and (3) cost. We explain below.

**Glitch Index Improvement** Suppose that the glitch index of the original data set  $D$  is  $G(D)$  as described in Sect. 5. After applying strategy  $S$ , we obtain a treated data set  $D^c$ , whose glitch index is  $G(D^c)$ . The glitch improvement in the data set  $D$  caused by the strategy  $S$  is  $\Delta_S(D)$ , given by

$$\Delta_S(D) = G(D^c) - G(D). \quad (4)$$

The data-cleaning process entails applying a sequence of strategies  $\{S_i\}$  until the final data set  $D^F$  meets the user’s specifications. Note that the sequence of  $\{\Delta_{S_i}(D)\}$  need not be monotonic. It is possible that a strategy might introduce more glitches, increasing the glitch index. For example, imputing missing values might introduce duplicates and outliers.

**Statistical Distortion** Dasu and Loh [7] propose the notion of *statistical distortion* to measure the effect of a cleaning strategy on the distributional properties of the original data  $D$ . The data are generated by a real-world process (e.g. traffic through a network router)  $P$ , and by cleaning the data, we are altering data values and distorting the statistical properties of the original data. When this statistical distortion is excessive, the cleaned data are significantly different from the original data and no longer represent the real-world process  $P$ . This could be the biggest data quality problem of all!

Statistical distortion measures the distance  $\mathcal{D}(D, D^F)$  of  $D$  from  $D^F$ . Distance between two data sets can be measured in many ways. Simple approaches include comparing means and other summary statistics of the two data sets. More sophisticated methods entail comparing multivariate distributions of the data sets using a histogram distance such as the Earth Mover Distance. See [7] for details.

**Cost Considerations** Along with glitch improvement and statistical distortion, cost  $C$  plays a major role in the choice of cleaning strategies. Every cleaning operation has a cost, whether it is computational or human (expert opinions). The user typically sets an upper bound  $K$  on cost, before the data-cleaning process is undertaken.

Therefore, a *candidate set of strategies*  $\mathcal{A}$  consists of strategies that meet the following criteria:

$$\begin{aligned} C &< K \\ G(D^F) &< r \\ \mathcal{D}(D, D^F) &< d \end{aligned}$$

where  $C$  represents the cost,  $K$  the maximum cost that the user is willing to incur, and  $r$  and  $d$  are tolerance bounds for the glitch index and statistical distortion, respectively. It is likely that there are multiple strategies that meet these criteria, in which case the user can choose the one that is most desirable among them (e.g., least expensive or least distorting).

## 7.1 An Experimental Framework

Note that it is often impractical to evaluate strategies by applying them to the data first. It defeats the very purpose of strategy selection. In the absence of other guidelines or expert opinion, strategies can be selected empirically using an experimental framework. The framework entails sampling from the data  $D$  and applying the strategies to multiple samples of data to evaluate them on the three criteria described above. See [7] for details.

## 8 Relevant Literature

We describe below a set of references that is neither complete nor exhaustive but is intended to serve as a starting point for further study.

Redman [13] provides an introductory overview of traditional definitions of data quality problems. Dasu and Johnson [5] discuss a comprehensive view of data quality and take a technical approach to the definition and cleaning of data glitches and include those based on process, constraint violation, and data interpretation.

Barnett and Lewis [1] discuss a statistical approach to outlier detection, while Chandola et al. survey the literature on anomaly detection from a computing perspective in [4]. Duplicate records and their definition, detection, and removal, are reviewed by Elmagarmid et al. in [8]. *Bellman*, discussed in [6], is a database browser for exploring glitches and glitch distributions in databases. Golab et al. [9] describe the use of functional dependencies to formulate constraints for identifying inconsistencies and to isolate subsets that match or violate the constraints. See [10] for a comprehensive overview of outlier detection methods.

The iterative nature and importance of measuring data quality after every cleaning pass is discussed by Berti-Equille and Dasu in [2]. The tutorial also provides an exhaustive introduction to recent advances in data quality and data quality mining.

Glitch signatures, patterns of glitches, glitch index, and glitch scoring are discussed in detail in [3].

A survey of data-cleaning methods can be found in [2].

In [7], the authors propose a three-pronged approach for evaluating data-cleaning strategies, including the novel concept of statistical distortion which measures the change in the statistical properties of the data caused by data cleaning. They also discuss an experimental framework for empirically choosing the best cleaning strategy as defined by their three-pronged approach.

## 9 Conclusion

In this chapter, we have presented a comprehensive data quality process that includes an overview of data glitches, statistical approaches to assessing data quality, and the development and evaluation of data-cleaning strategies. This is an active and exciting area of research given the sheer volume and variety of data types that are emerging as a consequence of the confluence of information, entertainment, and mobility. Open problems include glitch correlation, new metrics for data quality evaluation, glitch scoring, and statistical distortion [7].

## References

1. Barnett V, Lewis T (1994) Outliers in statistical data. Wiley, Chichester
2. Berti-Equille L, Dasu T (2009) Advances in data quality mining. Tutorial, KDD
3. Berti-Equille L, Dasu T, Srivastava D (2011) Discovery of complex glitch patterns: a novel approach to quantitative data cleaning. In: 2011 IEEE 27th international conference on data engineering (ICDE)
4. Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: a survey. ACM Comput Surv 41(3), Article 15, 58 p
5. Dasu T, Johnson T (2003) Exploratory data mining and data cleaning. Wiley, New York
6. Dasu T, Johnson T, Muthukrishnan S, Shkapenyuk V (2002) Mining database structure; or, how to build a data quality browser. In: Proceedings of the SIGMOD
7. Dasu T, Loh JM (2012) Statistical distortion: consequences of data cleaning. PVLDB 5(11):1674–1683
8. Elmagarmid AK, Ipeirotis PG, Verykios VS (2007) Duplicate record detection a survey. IEEE Trans Knowledge Data Eng 19(1):1–16
9. Golab L, Saha A, Karloff H, Srivastava D, Korn P (2009) Sequential dependencies. PVLDB 2(1):574–585
10. Kriegel H, Kroger P, Zimek A (2009) Outlier detection techniques. Tutorial, PAKDD
11. Liu X, Dong XL, Ooi BC, Srivastava D (2011) Online data fusion. PVLDB 4(11):932–943
12. Rao CR (1973) Linear statistical inference and its applications. Wiley, New York
13. Redman T (1997) Data quality for the information age. Artech House, Norwood

## **Part III**

# **Computational Aspects of Data Quality**

This part presents topics related to computational methods, tools and techniques required to meet data quality objectives. The contributions from the scientific research community, in particular from database research, are featured prominently in this part. Four specific topics have been targeted in this part, namely rules and constraints for data quality, addressing issues in record linkage (including duplicate detection and entity resolution), managing quality of uncertain data, and finally tracking data lineage and provenance.

The first topic relates to the traditional area of database integrity constraints. However, Leopoldo Bertossi and Loreto Bravo provide both a historical perspective and the current state of the art on generic and declarative approaches to data quality specification covering a range of rule and constraint classes for data consistency management and data repair.

The topic of record linkage is one of the most widely studied topics in data quality and data integration research (see the prologue of the handbook for details regarding literature on this topic). Record linkage has spawned a number of related topics such as similarity matching, duplicate detection, and entity resolution, to name a few. Hence there are two chapters dedicated to this topic. Pei Li and Andrea Maurino first present a chapter on record linking using a range of traditional and emerging computational techniques. Then, John Talburt and Yinle Zhou present a chapter devoted to entity resolution that focuses more on the practical adoption of various computational techniques for entity resolution within a master data management program.

The fourth chapter in this part is presented by Reynold Cheng and addresses the issue of dealing with data uncertainty, which is gaining importance due to the increase in volume and diversity of data from noisy sources such as GPS devices, sensor networks, and RFID readers.

The final chapter in this part is devoted to the topic of data (lineage) tracking and discovery of truth as data propagates through large information chains. This issue is pervasive in any application where data is integrated from multiple sources.

Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava address this issue through carefully crafted data fusion techniques that are capable to resolve conflicts in data received from multiple sources and thereby assist in improving the accuracy of truth discovery.

# Generic and Declarative Approaches to Data Quality Management

Leopoldo Bertossi and Loreto Bravo

**Abstract** Data quality assessment and data cleaning tasks have traditionally been addressed through procedural solutions. Most of the time, those solutions have been applicable to specific problems and domains. In the last few years we have seen the emergence of more generic solutions, and also of declarative and rule-based specifications of the intended solutions of data cleaning processes. In this chapter we review some of those recent developments.

## 1 Introduction

Data quality assessment and data cleaning have been mostly ad hoc, rigid, vertical, and application-dependent activities. There is a need for more general methodologies and solutions. Most of the existing approaches have been also procedural, provided in terms of specific mechanisms. However, their semantics and scope of applicability are not fully understood. Declarative approaches could be attempted in this regard. They specify, usually by means of a logic-based formalism, what is the intended result of a data cleaning process. The semantics of the specification tells us what the result should look like, if there are alternative solutions, and what are the conclusions that can be derived from the process. They also allow us, in principle, to better understand the range of applicability and complexity of the declaratively specified cleaning mechanism.

Considering the large body of literature accumulated in the areas of data quality assessment and cleaning, it is safe to say that there is a lack of fundamental research

---

L. Bertossi (✉)  
Carleton University, Ottawa, ON, Canada  
e-mail: [bertossi@scs.carleton.ca](mailto:bertossi@scs.carleton.ca)

L. Bravo  
Universidad de Concepción, Concepción, Chile  
e-mail: [lbravo@udec.cl](mailto:lbravo@udec.cl)

around the activities of data quality assessment and data cleaning. Fortunately, things are starting to change. Important impulses in this direction have come from two sources.

One of them is related to a new look at classic integrity constraints (ICs). Since they can be (and are) violated in many applications, they can still be used as desirable properties that could be enforced if necessary, cleaning the database from semantic anomalies. However, this enforcement could be only partial, full, or even virtual. It could also be imposed at query answering time, seeing ICs more like constraints on query answers than on database states (cf. Sect. 3).

The other source is related to the introduction of newer classes of ICs that are intended to capture data quality issues or conditions and are intended to be used to directly support data cleaning processes. We could call them *data quality constraints*. They have been proposed and investigated in the last few years and provide generic languages for expressing quality concerns [42]. They may be a suitable basis for declaratively specifying adaptive and generic data quality assessment and cleaning mechanisms (cf. Sects. 4.1 and 5.2).

Generic methods for data cleaning may be proposed for possibly solving a single but still general problem in the area, e.g., entity resolution, data editing (i.e., changes of data values in records), and incompleteness of data. They are abstract and parameterized approaches or mechanisms that can be applied to different specific instances of the data cleaning problem at hand, by instantiating parameters, methods, and modules as required by the specificity of the problem and application domain. For example, a general entity resolution algorithm may rely on matching functions to do the actual merging of records. In the abstract formulation, they are left rather open, except for some general requirements they have to satisfy. However, when the algorithm is applied to a specific domain, those matching functions become domain dependent, as expected.

Earlier generic methods in data cleaning were proposed in [52]. Actually, this is a framework for entity resolution whose aim is to separate the logic of data transformations from their actual implementations. Data transformations, e.g., matching and merging of records, are specified in an extension of SQL. The methodology was implemented and tested in the *AJAX* system [52] and inspired newer developments in the area, e.g., the *Swoosh* generic approach to entity resolution [7] (cf. Sect. 5.1). Other generic approaches are described in some articles in [39].

Declarative methods in data cleaning, again possibly for a specific and general problem, are expected to be based on a formal, say logic-based, specification of the intended results of the cleaning process. These results are usually represented by (or through) the models of the specification, which requires defining a precise semantics for the formalism at hand.

This chapter gives a survey of some recent research on generic and declarative methods in data cleaning. Considering that this is a vast area of research, we concentrate in more detail only on some problems. In Sect. 2 we review basics of integrity constraints in relational databases. In Sect. 3 we concentrate on *database repairs* with respect to classic semantic constraints. In Sect. 4, we present *conditional dependencies* for data quality assessment and data cleaning. In Sect. 5,

we demonstrate the value of declarative approaches to entity resolution (or deduplication), with emphasis on the use of *matching dependencies*. In Sect. 6, we make some final remarks and briefly mention some other problems and approaches as related to the previous sections. For more on data quality and data cleaning, we refer to the general reference [6] and to the earlier general survey [68].

## 2 Classic Integrity Constraints

Integrity constraints (ICs) are used to capture semantics of the outside world that is being modeled through the data model and the database. For this reason they are also called semantic constraints. ICs have been around at least since the inception of the relational model of data. Already in the classical and seminal papers in the area [36], it is possible to find the notions of integrity and consistency of a database.

ICs have been studied in general and have wide application in data management. A large body of research has been developed, in particular fundamental research has been carried out. Furthermore, methodologies for dealing with ICs are quite general and have broad applicability.

### 2.1 The Basics of ICs

A database can be seen as a model, i.e., as a simplified, abstract description, of an external reality. In the case of relational databases, one starts by choosing certain predicates of a prescribed arity. The *schema* of the database consists of this set of predicates, possibly *attributes*, which can be seen as names for the arguments of the predicates, together with an indication of the domains where the attributes can take their values. Having chosen the schema, the representation of the external reality is given in terms of relations, which are extensions for the predicates in the schema. This set of relations is called an *instance* of the schema.

For example, relational database for representing information about students of a university might be based on the schema consisting of the predicates *Students(StuNum, StName)* and *Enrollment(StuName, Course)*. The attribute *StuNum* is expected to take numerical values; *StuName*, character string values; and *Course*, alphanumeric string values. In Fig. 1, there is a possible instance for this schema.

In order to make the database a more accurate model of the university domain (or to be in a more accurate correspondence with it), certain conditions are imposed on the possible instances of the database. Those conditions are intended to capture more meaning from the outside application domain. In consequence, these conditions are called *semantic constraints* or *integrity constraints* (ICs). For example, a condition could be that, in every instance, the student name functionally depends upon the student number, i.e., a student number is assigned to at most

**Fig. 1** A database instance

Students		Enrollment	
StuNum	StuName	StuNum	Course
101	john bell	104	comp150
102	mary stein	101	comp100
104	claire stevens	101	comp200
107	pat norton	105	comp120

**Fig. 2** Another instance

Students		Enrollment	
StuNum	StuName	StuNum	Course
101	john bell	104	comp150
101	joe logan	101	comp100
104	claire stevens	101	comp200
107	pat norton		

one student name. This condition, called a *functional dependency* (FD), is denoted with  $StuNumber \rightarrow StuName$ , or  $Students : StuNumber \rightarrow StuName$ , to indicate that this dependency should hold for attributes of relation *Students*. Actually, in this case, since all the attributes in the relation functionally depend on *StuNum*, the FD is called a *key constraint*.

Integrity constraints can be declared together with the schema, indicating that the instances for the schema should all satisfy the integrity constraints. For example, if the functional dependency  $Students : StuNumber \rightarrow StuName$  is added to the schema, the instance in Fig. 1 is consistent, because it satisfies the FD. However, the instance in Fig. 2 is *inconsistent*. This is because this instance does not satisfy, or, what is the same, violates the functional dependency (the student number 101 is assigned to two different student names).

It is also possible to consider with the schema a *referential integrity constraint* that requires that every student (number) in the relation *Enrollment* appears, associated with a student name, in relation *Students*, the official “table” of students. This is denoted with  $Enrollment[StuNum] \subseteq Students[StuNum]$  and is a form of *inclusion dependency*. If this IC is considered in the schema, the instance in Fig. 1 is inconsistent, because student 105 does not appear in relation *Students*. However, if only this referential constraint were associated to the schema, the instance in Fig. 2 would be consistent. The combination of the given referential constraint and the functional dependency creates a *foreign key constraint*: The values for attribute *StuNum* in *Enrollment* must appear in relation *Students* as values for its attribute *StuNum*, and this attribute form a *key* for *Students*.

It can be seen that the notion of consistency is relative to a set of integrity constraints. A database instance that satisfies each of the constraints in this set is said to be *consistent* and *inconsistent* otherwise.

The two particular kinds of integrity constraints presented above and also other forms of ICs can be easily expressed in the language of predicate logic. For example, the FD above can be expressed by the symbolic sentence

$$\forall x \forall y \forall z ((Students(x, y) \wedge Students(x, z)) \rightarrow y = z), \quad (1)$$

whereas the referential constraint above can be expressed by

$$\forall x \forall y (\text{Enrollment}(x, y) \longrightarrow \exists z \text{Students}(x, z)). \quad (2)$$

Notice that this language of predicate logic is determined by the database schema, whose predicates are now being used to write down logical formulas. We may also use “built-in” predicates, like the equality predicate. Thus, ICs can be seen as forming a set  $\Sigma$  of sentences written in a language of predicate logic.

A database instance can be seen as an *interpretation structure*  $D$  for the language of predicate logic that is used to express ICs. This is because an instance has an underlying domain and (finite) extensions for the predicates in the schema. Having the database instance as an interpretation structure and the set of ICs as a set of symbolic sentences makes it possible to simply apply the notion of satisfaction of a formula by a structure of first-order predicate logic. In this way, the notion of satisfaction of an integrity constraint by a database instance is a precisely defined notion: the database instance  $D$  is consistent if and only if it satisfies  $\Sigma$ , which is commonly denoted with  $D \models \Sigma$ .

Since it is usually assumed that the set of ICs is consistent as a set of logical sentences, in databases the notion of consistency becomes a condition on the database instance. Thus, this use of the term “consistency” differs from its use in logic, where consistency characterizes a set of formulas.

## 2.2 Checking and Enforcing ICs

Inconsistency is an undesirable property for a database. In consequence, one attempts to keep it consistent as it is subject to updates. There are a few ways to achieve this goal. One of them consists in declaring the ICs together with the schema, and the database management system (DBMS) will take care of the *database maintenance*, i.e., of keeping it consistent. This is done by rejecting transactions that may lead to a violation of the ICs. For example, the DBMS should reject the insertion of the tuple  $(101, \text{sue jones})$  into the instance in Fig. 1 if the FD (1) was declared with the schema (as a key constraint). Unfortunately, the classes of ICs for which most commercial DBMSs offer this kind of automated, built-in support are quite restricted [72].

An alternative way of keeping consistency is based on the use of triggers (or active rules) that are stored in the database [32]. The reaction to a potential violation is programmed as the action of the trigger: if a violation is about to be produced or is produced, the trigger automatically reacts, and its action may reject the violating transaction or compensate it with additional updates, to make sure that at the end, consistency is reestablished. Consistency can also be enforced through the application programs that interact with the DBMS. However, the correctness of triggers or application programs with respect to ensuring database consistency is not guaranteed by the DBMS.

Inconsistency of a DB under updates can be checked via *violation views* that transitorily store violating tuples, if any. IC satisfaction corresponds to an empty violation view. Some consistency restoration policy can be applied on the basis of the violation views. No wonder that database maintenance and *view maintenance* [60] are closely related problems. It is possible to apply similar techniques to both problems. For example, to check IC violation and view freshness, i.e., the correspondence with the base tables, inductive incremental techniques can be applied. More precisely, on the basis of the syntactic form of an IC or a view definition, it is possible to identify the updates that are relevant under the assumption that the database is consistent or the view is up-to-date when those updates are executed. Potential IC violations or changes in view extensions are checked only for those relevant updates [16, 67].

It is the case that, for whatever reasons, databases may become inconsistent, i.e., they may violate certain ICs that are considered to be relevant to maintain for a certain application domain. This can be due to several reasons, e.g., poorly designed or implemented applications that fail to maintain the consistency of the database, or ICs for which a DBMS does not offer any kind of support, or ICs that are not enforced for better performance of application programs or DBMSs, or ICs that are just assumed to be satisfied based on knowledge about the application domain and the kind of updates on the database. It is also possible to have a legacy database on which semantic constraints have to be imposed, or more generally, a database on which imposing new constraints depending on specific needs, e.g., user constraints, becomes necessary.

In the area of data integration the satisfaction of desirable ICs by a database is much more difficult to achieve. One can have different autonomous databases that are separately consistent with respect to their own, local ICs. However, when their data are integrated into a single database, either material or virtual, certain desirable global ICs may not be satisfied. For example, two university databases may use the same numbers for students. If their data are put together into an integrated database, a student number might be assigned to two different students (cf. Sect. 3.3).

When trying to use an inconsistent database, the application of some *data cleaning* techniques may be attempted, to cleanse the database from data that participate in the violation of the ICs. This is done sometimes. However, data cleaning is a complex and nondeterministic process, and it may also lead to the loss of information that might be useful. Furthermore, in certain cases like virtual data integration, where the data stay at the autonomous data sources, there is no way to change the data without the ownership of the sources.

One might try to live with an inconsistent database. Actually, most likely one will be forced to keep using it, because there is still useful information in it. It is also likely that most of the information in it is somehow consistent. Thus, the challenge consists in retrieving from the database only information that is consistent. For example, one could pose queries to the database at hand but expecting to obtain only answers that are semantically correct, i.e., which are consistent with the ICs. This is the problem of *consistent query answering* (CQA), which can be formulated in general terms as the one of characterizing and computing semantically correct answers to queries posed to inconsistent databases [2].

**Fig. 3** Two repairs of *Students* in Fig. 2

<i>Students</i>		<i>Students</i>	
<i>StuNum</i>	<i>StuName</i>	<i>StuNum</i>	<i>StuName</i>
101	john bell	101	joe logan
104	claire stevens	104	claire stevens
107	pat norton	107	pat norton

### 3 Repairs and Consistent Answers

The notion of consistency of a database is a holistic notion that applies to the entire database, and not to portions of it. In consequence, in order to pursue this idea of retrieving consistent query answers, it becomes necessary to characterize the consistent data in an inconsistent database first. The idea that was proposed in [2] is as follows: the consistent data in an inconsistent data are the data that are invariant under all possible way of restoring the consistency by performing minimal changes on the initial database. That is, no matter what minimal consistency restoration process is applied to the database, the consistent data stay in the database. Each of the consistent versions of the original instance obtained by minimal changes is called a *minimal repair*, or, simply, a *repair*.

It becomes necessary to be more precise about the meaning of minimal change. In between, a few notions have been proposed and studied (cf. [9, 10, 35] for surveys of CQA). Which notion to use may depend on the application. The notion of minimal change can be illustrated using the definition of repair given in [2]. First of all, a database instance  $D$  can be seen as a finite set of ground atoms (or database tuples) of the form  $P(\bar{c})$ , where  $P$  is a predicate in the schema and  $\bar{c}$  is a finite sequence of constants in the database domain. For example,  $Students(101, john\ bell)$  is an atom in the database. Next, it is possible to compare the original database instance  $D$  with any other database instance  $D'$  (of the same schema) through their symmetric difference  $D \Delta D' = \{A \mid A \in (D \setminus D') \cup (D' \setminus D)\}$ .

Now, a repair of an instance  $D$  with respect to a set of ICs  $\Sigma$  is defined as an instance  $D'$  that is consistent, i.e.,  $D' \models \Sigma$ , and for which there is no other consistent instance  $D''$  that is closer to  $D$  than  $D'$ , i.e., for which it holds  $D \Delta D'' \subsetneq D \Delta D'$ . For example, the database in Fig. 2 has two repairs with respect to the FD (1). They are shown in Fig. 3 and are obtained each by deleting one of the two conflicting tuples in relation *Students* (relation *Enrollment* does not change).

Having defined the notion of repair, a *consistent answer* from an instance  $D$  to a query  $\mathcal{Q}(\bar{x})$  with respect to a set  $\Sigma$  of ICs is defined as an answer  $\bar{c}$  to  $\mathcal{Q}$  that is obtained from every possible repair of  $D$  with respect to  $\Sigma$ . That is, if the query  $\mathcal{Q}$  is posed to each of the repairs,  $\bar{c}$  will be returned as a usual answer to  $\mathcal{Q}$  from each of them.

For example, if the query  $\mathcal{Q}_1(x, y) : Students(x, y)$ , asking for the tuples in relation *Students*, is posed to the instance in Fig. 2, then  $(104, claire\ stevens)$  and  $(107, pat\ norton)$  should be the only consistent answers with respect to the FD (1). Those are the tuples that are shared by the extensions of *Students* in the two repairs.

Now, for the query  $\mathcal{Q}_2(x) : \exists y Students(x, y)$ , i.e., the projection on the first attribute of relation *Students*, the consistent answers are (101), (104), and (107).

There might be a large number of repairs for an inconsistent database. In consequence, it is desirable to come up with computational methodologies to retrieve consistent answers that use only the original database, in spite of its inconsistency. Such a methodology, that works for particular syntactic classes of queries and ICs, was proposed in [2]. The idea is to take the original query  $\mathcal{Q}$  that expects consistent answers, and syntactically transform it into a new query  $\mathcal{Q}'$ , such that the *rewritten query*  $\mathcal{Q}'$ , when posed to the original database, obtains as usual answers the consistent answers to query  $\mathcal{Q}$ . The essential question is, depending on the language in which  $\mathcal{Q}$  is expressed, what kind of language is necessary for expressing the rewriting  $\mathcal{Q}'$ . The answer to this question should also depend on the kind of ICs being considered.

The idea behind the rewriting approach presented in [2] can be illustrated by means of an example. The consistent answers to the query  $\mathcal{Q}_1(x, y) : Students(x, y)$  above with respect to the FD (1) can be obtained by posing the query  $\mathcal{Q}'(x, y) : Students(x, y) \wedge \neg \exists z (Students(x, z) \wedge z \neq y)$  to the database. The new query collects as normal answers those tuples where the value of the first attribute is not associated to two different values of the second attribute in the relation. It can be seen that the set of answers for the new query can be computed in polynomial time in the size of the database.

In this example, a query expressed in first-order predicate logic was rewritten into a new query expressed in the same language. It has been established in the literature that, for complexity-theoretic reasons, a more expressive language to do the rewriting of a first-order query may be necessary. For example, it may be necessary to do the rewritings as queries written in expressive extensions of Datalog [3, 5, 40, 59]; see Sect. 3.1 for more details.

If a database is inconsistent with respect to referential ICs, like the instance in Fig. 1 and the constraint in (2), it is natural to restore consistency by deleting tuples or inserting tuples containing *null values* for the existentially quantified variables in the ICs. For example, the tuple (105, comp120) could be deleted from *Enrollment*, or the tuple (105, null) could be inserted in relation *Students*. This requires a modification of the notion of repair and a precise semantics for satisfaction of ICs in the presence of null values [21, 30].

Some repair semantics consider changes of attribute values as admissible basic repair actions [12, 18, 50, 51, 74], which is closer to many data cleaning processes. Usually, it is the number of these changes that is minimized. With a change of repair semantics, the problems of complexity and computation of consistent answers, query rewriting, and also of specification of database repairs have to be reinvestigated.

CQA involves, possibly only implicitly, the whole class of database repairs. However, some research in this area, and much in the spirit of classical data cleaning, has also addressed the problem of computing a single “good” repair [18, 37, 75], or a single universal repair that can act for some tasks as a good representative of the class of repairs [71], or an approximate repair [12, 61].

### 3.1 Answer Set Programs for Database Repairing

ICs like (1) and (2) specify desirable properties of a database. However, when they are not satisfied by a given instance, they do not tell us how to change the instance so that they hold again. This requires a separate specification of the repair process. A declarative specification of it will tell us, in a language with a clear logical semantics, what are the intended results of the repair process, without having to explicitly express how to go about computing them. The intended models of the specification should correspond to the expected results.

In an ideal situation, the declarative specification can also be used as (the basis for) an *executable specification* that can be used, for example, for *computing* the models (the repairs in our case) or computing query answers from the specification (the consistent answers in our case).

Actually, it turns out that the class of repairs of an inconsistent database can be specified by means of disjunctive logic programs with stable model semantics, aka *answer-set programs* [24]. The programs can be modified in order to accommodate computation of single repairs or an approximation to one of them, enabling in this way a form of rule-based data cleaning process, in this case of restoration of semantic integrity. In this section we briefly describe the approach proposed in [5, 21] (cf. also [30] for more details).

The idea behind *repair programs* is to represent the process of capturing and resolving inconsistencies through rules and enforce the minimality of repairs through the minimality of the stable models of the program.

A repair program is produced for a database schema, including a non-necessarily enforced set of ICs. More specifically, for each database predicate  $P$ , a new predicate  $\bar{P}$  is introduced. It corresponds to  $P$  augmented with an extra attribute that can take the following values (annotation constants):  $t$ ,  $f$ ,  $t^*$ , and  $t^{**}$ , whose intended semantics is as follows.

Atom  $\bar{P}(\bar{t}, t)$  (respectively,  $\bar{P}(\bar{t}, f)$ ) indicates the insertion of tuple  $\bar{t}$  into relation  $P$  (respectively, the deletion of  $\bar{t}$  from  $P$ ) during the repair process. Atom  $\bar{P}(\bar{t}, t^*)$  indicates that tuple  $\bar{t}$  was in the original extension of  $P$  or is inserted in the repair process. Finally, atom  $\bar{P}(\bar{t}, t^{**})$  indicates that tuple  $\bar{t}$  belongs to the final extension of  $P$ . For example, if a stable model of the repair program for the FD  $StuNumber \rightarrow StuName$  contains the atoms  $Students(101, john\ bell)$ ,  $Students(101, joe\ logan)$ ,  $Students_(101, john\ bell, f)$ , and  $Students_(101, joe\ logan, t^{**})$ , it means that tuple  $Students(101, john\ bell)$  was removed in order to solve an inconsistency, and tuple  $Students(101, joe\ logan)$  belongs to a repair.

As an example, the repair program for the student database with the FD (1) and the referential constraint (2) contains the original database tuples as program facts, plus:

1. A rule to enforce the FD:

$$Students_-(x, y, f) \vee Students_-(x, z, f) \leftarrow Students_-(x, y, t^*), Students_-(x, z, t^*), \\ y \neq z, x \neq null, y \neq null, z \neq null.$$

The right-hand side of the rule checks if there are tuples (that were originally part of the database or that were made true while repairing) that violate the FD. If that is the case, it solves the inconsistency by removing one of the two tuples (the left-hand side of the rule). Notice that there can be a violation of the FD in a database with *null* only if  $x$ ,  $y$ , and  $z$  are not *null*.

## 2. Rule to enforce the referential constraint:

$$\begin{aligned} \text{Aux}(x) &\leftarrow \text{Students}_\perp(x, z, \mathbf{t}^*), \text{not } \text{Students}_\perp(x, z, \mathbf{f}), x \neq \text{null}, z \neq \text{null}. \\ \text{Enrollment}_\perp(x, y, \mathbf{f}) \vee \text{Students}_\perp(x, \text{null}, \mathbf{t}) &\leftarrow \text{Enrollment}_\perp(x, y, \mathbf{t}), \text{not } \text{Aux}(x), x \neq \text{null}. \end{aligned}$$

The first rule populates an auxiliary predicate storing all the students numbers stored in the table *Students* that are not deleted in the repair process. The second rule checks through its right-hand side if for any enrolled student, his/her number does appear in the table *Students*. If there is a violation, to solve the inconsistency, it forces either to remove the tuple from *Enrollment* or add a tuple to *Students*.

## 3. Rules defining the annotation semantics:

$$\begin{aligned} \text{Students}(x, y, \mathbf{t}^*) &\leftarrow \text{Students}(x, y). \\ \text{Students}(x, y, \mathbf{t}^*) &\leftarrow \text{Students}_\perp(x, y, \mathbf{t}). \\ \text{Students}(x, y, \mathbf{t}^{**}) &\leftarrow \text{Students}_\perp(x, y, \mathbf{t}^*), \text{not } \text{Students}_\perp(x, y, \mathbf{f}). \\ \left. \begin{aligned} &\leftarrow \text{Students}_\perp(x, y, \mathbf{t}), \text{Students}_\perp(x, y, \mathbf{f}). \end{aligned} \right\} \begin{array}{l} \text{Similarly for} \\ \text{Enrollment} \end{array} \end{aligned}$$

The first two rules ensure that tuples with  $\mathbf{t}^*$  are those in the original database or those that are made true through the repair. The third rules collect tuples that belong to a final repair. The last rule is a *program constraint* that discards models where a tuple is both made true and false.

If rules 1–3 are combined with the database instance in Fig. 2, the repair program has two stable models  $\mathcal{M}_1$  and  $\mathcal{M}_2$  such that in  $\mathcal{M}_1$  predicate  $\text{Students}_\perp = \{(101, john\ bell, \mathbf{t}^*), (101, joe\ logan, \mathbf{t}^*), (104, claire\ stevens, \mathbf{t}^*), (101, john\ bell, \mathbf{f}), (101, joe\ logan, \mathbf{t}^{**}), (104, claire\ stevens, \mathbf{t}^{**}), (107, pat\ norton, \mathbf{t}^{**})\}$ , whereas in  $\mathcal{M}_2$  predicate  $\text{Students}_\perp = \{(101, john\ bell, \mathbf{t}^*), (101, joe\ logan, \mathbf{t}^*), (104, claire\ stevens, \mathbf{t}^*), (101, joe\ logan, \mathbf{f}), (101, john\ bell, \mathbf{t}^{**}), (104, claire\ stevens, \mathbf{t}^{**}), (107, pat\ norton, \mathbf{t}^{**})\}$ . These models correspond to the repairs shown in Fig. 3.

Given a database  $D$  and a set of ICs  $\Sigma$ , if  $\Sigma$  is *RIC-acyclic* [21], i.e., there is no cycle through referential constraints, then there is a one-to-one correspondence between repairs and stable models of the program: the tuples annotated with  $\mathbf{t}^{**}$  in a stable model form a repair of  $D$  with respect to  $\Sigma$ .

Consistent query answers can be computed directly from the repair program using the *cautious* or *skeptical semantics*, according to which an atom is true if it belongs to all models of the program. For example, if we want to pose the query  $\mathcal{Q}_1(x, y) : \text{Students}(x, y)$  or the query  $\mathcal{Q}_2(x) : \exists y \text{Students}(x, y)$ , we would simply add the rule  $\text{Ans}_{\mathcal{Q}_1}(x, y) \leftarrow \text{Student}_\perp(x, y, \mathbf{t}^{**})$  to the program, or  $\text{Ans}_{\mathcal{Q}_2}(x) \leftarrow \text{Student}_\perp(x, y, \mathbf{t}^{**})$ , respectively. Cautious reasoning from the extended program returns the consistent query answers to the queries.

There are techniques to improve the efficiency of repair programs by concentrating only on what is relevant for the query at hand [30]. Repair programs can also be modified to capture other semantics. For example, with *weak constraints*, one can specify repairs that minimize the *number* of tuple insertions/deletions [3].

### 3.2 Active Integrity Constraints

When specifying a database that could possibly not satisfy certain ICs, we might want to explicitly indicate how to restore consistency when an IC is violated. In this way, we specify both the constraint and its enforcement. In this direction, in [31], *active integrity constraints* were proposed. They extend traditional ICs with specifications of the actions to perform in order to restore their satisfaction, declaratively bringing ICs and active rules together.

For example, if we wanted to repair violations of the constraint (2), by only removing tuples from *Enrollment*, we could specify the following active IC:

$$\forall x \forall y [\text{Enrollment}(x, y), \neg \exists z \text{Students}(x, z) \supset \neg \text{Enrollment}(x, y)],$$

with the effect of removing a tuple *Enrollment* when it participates in a violation.

If, instead, we want to repair by either removing or inserting, we could use  $\forall x \forall y [\text{Enrollment}(x, y), \neg \exists z \text{Students}(x, z) \supset \neg \text{Enrollment}(x, y) \vee +\text{Students}(x, \perp)]$ , where  $\perp$  is a constant denoting an *unknown* value.

A *founded repair* is a minimal repair (as defined in Sect. 3) where each insertion and removal is supported by an active IC. The founded repairs can be computed using logic programs with stable models semantics (cf. [31] for details).

### 3.3 ICs and Virtual Data Integration

Virtual data integration [11, 62] is about providing a unified view of data stored in different sources through what is usually called a *mediator*. This is achieved by the definition of a global database schema and mappings between this schema and the source schemas. Queries that are received by the integration system via the mediator are automatically rewritten into a set of sub-queries that are sent to the sources. The answers from the sources are combined to give an answer to the user query. In this setting, specially because of the autonomy of the sources, it is hard or impossible to ensure that data are consistent across the different sources. Actually, it is natural to try to impose *global ICs*, i.e., on the global schema, with the intention to capture the semantics of the integration system as a whole. However, there is nothing like an explicit global database instance on which these ICs can be easily checked or enforced.

Consider, for example, a university that uses a virtual data integration system that integrates the departmental databases, which are maintained by the departments. The university officials can pose queries through a global schema that logically unifies the data, without the need of transforming and moving all the data into a single place.

**Fig. 4** Two data sources with student information from different departments

<i>StuPs</i>		<i>StuCS</i>		
<i>StuNum</i>	<i>StuName</i>	<i>StuNum</i>	<i>StuName</i>	<i>Admission</i>
101	john bell	104	alex pelt	2011
104	claire stevens	121	ana jones	2012
107	pat norton	137	lisa reeves	2012

For simplicity, assume we have only two data sources, containing information about psychology and computer science students, in tables *StuPs* and *StuCS*, respectively, as shown in Fig. 4. The mediator offers a global schema

$$\textit{StuUniv}(\textit{StuNum}, \textit{StuName}, \textit{Admission}, \textit{Department}).$$

The relationship between the sources and the global schema can be defined through the following mappings:

$$\textit{StuPs}(x, y) \leftarrow \textit{StuUniv}(x, y, z, "psychology"). \quad (3)$$

$$\textit{StuCS}(x, y, z) \leftarrow \textit{StuUniv}(x, y, z, "comp. sci."). \quad (4)$$

These mappings are defined according to the *local-as-view* (LAV) approach, where the sources are defined as views over the global schema.<sup>1</sup> Usually sources are assumed to be *open* (or *incomplete*), in the sense that these mappings require that potential global instances, i.e., for the global schema, if we decided to built and materialize them, have to contain *at least* the data that are needed to reconstruct the source contents through the view definitions (3) and (4). For example, the first mapping requires that, for every tuple in *StuPs*, there exists a tuple in *StuUniv* with a value for *Admission* (possibly not known) and with the value *psychology* for attribute *Department*.

As a consequence, a global instance satisfies a mapping if the result of applying the mapping to it (through the right-hand side of the mapping) produces a superset of the data in the source that is defined by it. If a global instance satisfies all the mappings, it is called a *legal instance* of the integration system.

The semantics of query answering in such a data integration system is the one of *certain answers*. More precisely, an answer to a query expressed in terms of the global schema is a certain answer and it is an (usual) answer from each of the possible legal instances. When dealing with monotone queries, say without negation, it is enough to concentrate on the *minimal legal instances*. They are the legal instances that are minimal under set inclusion.

In the university database the minimal legal instances can be represented by the instance in Fig. 5 where the variables can take any value in the domain. A legal instance will be any instance of the global schema that is a superset of one of these minimal legal instances. The set of certain answers to query  $\mathcal{Q}_3(x) : \exists y \exists z \exists w \textit{Students}(x, y, z, w)$  is  $\{101, 104, 107, 121, 137\}$  since its elements belong to all the minimal legal instances. Query  $\mathcal{Q}_4(x, z) : \exists y \exists w \textit{Students}(x, y, z, w)$  will

---

<sup>1</sup>Cf. [62] for alternative approaches to mapping definitions.

**Fig. 5** A minimal legal instance with **X**, **Y**, and **Z** representing arbitrary values from the domain

<i>StuUniv</i>			
<i>StuNum</i>	<i>StuName</i>	<i>Admission</i>	<i>Department</i>
101	john bell	<b>X</b>	psychology
104	claire stevens	<b>Y</b>	psychology
107	pat norton	<b>Z</b>	psychology
104	alex pelt	2011	comp. sci.
121	ana jones	2012	comp. sci.
137	lisa reves	2012	comp. sci.

return only  $\{(104, 2011), (121, 2012), (137, 2012)\}$  since different minimal legal instances will return different admission years for the students of the psychology department.

It is possible to specify this class of legal instances as the models of an answer-set program [11], and certain answers can be computed by cautious reasoning from the program. In the university example, the program that specifies the minimal legal instances contains the source facts, plus the rules:

1.  $\text{dom}(\mathbf{X})$ . (for every value **X** in the domain)
2.  $\text{StuUniv}(x, y, z, \text{"psychology"}) \leftarrow \text{StuPs}(x, y), F_1(x, y, z).$   
 $\text{StuUniv}(x, y, z, \text{"comp. sci."}) \leftarrow \text{StuPs}(x, y, z).$
3.  $F_1(x, y, z) \leftarrow \text{StuPs}(x, y), \text{dom}(z), \text{choice}((x, y), z).$

The first rule adds all the elements of the finite domain.<sup>2</sup> The next two rules compute the minimal legal instances from the sources. Since the psychology department does not provide information about the admission year of the student, a value of the domain needs to be added to each instance. This is achieved with predicate  $F_1(X, Y, Z)$ , which satisfies the functional dependency  $X, Y \rightarrow Z$  and assigns in each model a different value for  $Z$  for each combination of values for  $X$  and  $Y$ . This requirement for  $F_1$  is enforced with the `choice` operator in the last rule, whose semantics can be defined by ordinary rules with a stable models semantics [56]. For more details, more complex cases, and examples, see [11].

So far, we have not considered global ICs. IC violations are likely in integration systems, and dealing with global ICs is a real challenge. In our ongoing example, we want to enforce that the *StuNum* is unique within the university. Both data sources in Fig. 4 satisfy this requirement, but when we combine the data, every legal instance will violate this constraint since they will contain students Claire Stevens and Alex Pelt with the same student number. As in the case of traditional relational databases, most of the data are still consistent, and we would like to be able to still provide answers for the consistent data.

Different approaches have been explored for dealing with global ICs. One of them consists in applying a repair semantics as in Sect. 3. More precisely, if a minimal legal instance does not satisfy a set of global ICs, it is repaired as before. The consistent answers to global queries are those that can be obtained from every

---

<sup>2</sup>For details of how to treat infinite domains see [11].

StuUniv				StuUniv			
StuNum	StuName	Admission	Department	StuNum	StuName	Admission	Department
101	john bell	X	psychology	101	john bell	X	psychology
104	claire stevens	Y	psychology	107	pat norton	Z	psychology
107	pat norton	Z	psychology	104	alex pelt	2011	comp. sci.
121	ana jones	2012	comp. sci.	121	ana jones	2012	comp. sci.
137	lisa reeves	2012	comp. sci.	137	lisa reeves	2012	comp. sci.

**Fig. 6** Repairs of the global integration system

repair from every minimal legal instance [11]. In our example, the (global) repairs in Fig. 6 are obtained by deleting either  $(104, \text{claire stevens}, Y, \text{psychology})$  or  $(104, \text{alex pelt}, 2011, \text{comp. sci.})$  from every minimal legal instance in Fig. 5. Now, the consistent answers to query  $\mathcal{Q}_3$  will coincide with the certain answers since 104 is also part of every repair. On the other hand, the consistent answers to query  $\mathcal{Q}_4$  now do not contain  $(104, 2011)$  since that answer is not obtained from all repairs.

Notice that the program above that specifies the minimal legal instances can be then extended with the rules introduced in Sect. 3.1, to specify the repairs of the minimal legal instances with respect to the global FD [11]. Next, in order to retrieve consistent answers from the integration system, the already combined program can be further extended with the query program.

Since the repair process may not respect the *nature of the sources*, in terms of being open, closed, etc. (as it is the case with the global repairs we just showed), we may decide to ignore this aspect [11] or, whenever possible, go for a repair semantics that respects this nature. For example, inserting global tuples may never damage the openness of a source [26, 27]. However, if these tuple insertions are due to the enforcement of inclusion dependencies, the presence of functional dependencies may create a problem since those insertions might violate them. This requires imposing some conditions on the syntactic interaction of FDs and inclusion dependencies [26].

Another possibility is to conceive a mediated system as an ontology that acts as a metadata layer on top of incomplete data sources [63]. The underlying data can be *chased* by means of the mappings and global ICs, producing data at the global level and extending the original extensional data [28].

Under any of these approaches, the emphasis is on certain query answering, and a full computation of legal instances, repairs, or chase extensions should be avoided whenever possible.

## 4 Data Dependencies and Data Quality

In Sect. 2 we have considered techniques for data quality centered in classical ICs, such as functional dependencies and referential constraints. These constraints, however, are not always expressive enough to represent the relationships among values for different attributes in a table.

<i>Staff</i>							<i>Journalist</i>	
<i>Num</i>	<i>Name</i>	<i>Type</i>	<i>Street</i>	<i>City</i>	<i>Country</i>	<i>Zip</i>	<i>SNum</i>	<i>Role</i>
01	john	admin	First	Miami	US	33114	01	news
02	bill	journalist	Bronson	Miami	US	33114	02	news
03	nick	journalist	Glebe	Tampa	US	33605	02	columnist
04	ana	admin	Crow	Glasgow	UK	G11 7HS	03	columnist
05	mary	journalist	Main	Glasgow	UK	G11 7HS		

**Fig. 7** The *Newspaper* database

Let us consider the example in Fig. 7 of a database storing information of the staff working at a newspaper. There are several restrictions on the data that can be represented using classical constraints. For example, in table *Staff* we have two FDs:  $(Num \rightarrow Name, Type, Street, City, Country, Zip)$  and  $(Zip, Country \rightarrow City)$ . We also have the inclusion dependency:  $(Journalist[Num] \subseteq Staff[num])$ . It is easy to check that the instance in Fig. 7 satisfies all these constraints.

The data, however, is not consistent with respect to other dependencies that cannot be expressed using this type of constraints since they apply only to some tuples in the relation. For example, in the UK, the zip code uniquely determines the street. If we try to enforce this requirement using the FD  $Zip \rightarrow Street$ , we would not obtain what we want since this requirement would be imposed also for tuples corresponding to the US. Also, we cannot express that every journalist in *Staff* should have a role assigned in table *Journalist* and that every *Num* in *Journalist* belongs to a tuple in table *Staff* with the  $SNum = Num$  and  $Type = "journalist"$ . The given instance does not satisfy any of these additional constraints. In order to be able to clean the data, we need to consider those dependencies that apply only under certain conditions.

## 4.1 Conditional Dependencies

In [19, 22], *conditional functional dependencies* and *conditional inclusion dependencies* are introduced, to represent data dependencies that apply only to tuples satisfying certain conditions.

A conditional functional dependency (CFD) is a pair  $(X \rightarrow Y, T_p)$ , where  $X \rightarrow Y$  is a classical functional dependency and  $T_p$  is a *pattern tableau*, showing attributes among those in  $X$  and  $Y$ . For every attribute  $A$  of  $T_p$  and every tuple  $t_p \in T_p$ , it holds that  $t_p[A]$  is a constant in the domain of  $A$  or an unnamed variable “\_”.

For the newspaper database we could define a CFD for relation *Staff* that enforces that for the UK, the zip code determines both the city and the street:

$$\psi_1 = (Country, Zip \rightarrow Street, City, T_1), \quad \text{with } T_1: \begin{array}{|c|c|c|c|} \hline Country & Zip & Street & City \\ \hline UK & - & - & - \\ \hline \end{array}$$

Two data values  $n_1$  and  $n_2$  match, denoted  $n_1 \asymp n_2$  if  $n_1 = n_2$  or one of  $n_1, n_2$  is “ $\perp$ ”. Two tuples match, denoted  $t_1 \asymp t_2$ , if they match componentwise. Now, a CFD ( $X \rightarrow Y, T_p$ ) is satisfied by a database instance if for every  $t_p \in T_p$  and every pair of tuples  $t_1$  and  $t_2$  in the database, if  $t_1[X] = t_2[X] \asymp t_p[X]$ , then  $t_1[Y] = t_2[Y] \asymp t_p[Y]$ .

Constraint  $\psi_1$  is violated by the instance in Fig. 7 since the last two tuples are from the UK; they share the same zip code but they are associated to different streets.

A conditional inclusion dependency (CIND) is a pair  $(R_1[X; X_p] \subseteq R_2[Y; Y_p], T_p)$  where:

- (a)  $R_1$  and  $R_2$  are database predicates
- (b)  $X, X_p$  (respectively,  $Y$  and  $Y_p$ ) are disjoint lists of attributes of  $R_1$  (respectively,  $R_2$ ),<sup>3</sup>
- (c)  $T_p$  is a pattern tableau that contains all the attributes in  $X, X_p, Y$  and  $Y_p$ , and  $t_p[X] = t_p[Y]$ .

In this case, the inclusion dependency  $R_1[X] \subseteq R_2[Y]$  is said to be embedded in the CIND.

For the newspaper database we could define two CINDs:

$\psi_2 = (Staff[Num; Type] \subseteq Journalist[SNum; nil], T_2)$ , with $T_2$ :	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="text-align: center;"><i>Num</i></td><td style="text-align: center;"><i>Type</i></td><td style="border-left: 1px solid black; text-align: center;"><i>SNum</i></td></tr> <tr> <td style="text-align: center;">—</td><td style="text-align: center;">Journalist</td><td style="border-left: 1px solid black; text-align: center;">—</td></tr> </table>	<i>Num</i>	<i>Type</i>	<i>SNum</i>	—	Journalist	—
<i>Num</i>	<i>Type</i>	<i>SNum</i>					
—	Journalist	—					
$\psi_3 = (Journalist[SNum; nil] \subseteq Staff[Num; Type], T_3)$ , with $T_3$ :	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="text-align: center;"><i>SNum</i></td><td style="text-align: center;"><i>Num</i></td><td style="text-align: center;"><i>Type</i></td></tr> <tr> <td style="text-align: center;">—</td><td style="text-align: center;">—</td><td style="text-align: center;">Journalist</td></tr> </table>	<i>SNum</i>	<i>Num</i>	<i>Type</i>	—	—	Journalist
<i>SNum</i>	<i>Num</i>	<i>Type</i>					
—	—	Journalist					

Constraint  $\psi_2$  enforces that every *Num* of a journalist in relation *Staff* has a role assigned to it in relation *Journalist*. Constraint  $\psi_3$  enforces that in every *SNum* in relation *Staff* there is a tuple with the same number *Num* of type *Journalist*.

A CIND  $(R_1[X; X_p] \subseteq R_2[Y; Y_p], T_p)$  is satisfied, if for every tuple  $t_1 \in R_1$  and every  $t_p \in T_p$  if  $t_1[X, X_p] \asymp t_p[X, X_p]$  then there exists a tuple  $t_2 \in R_2$  such that  $t_2[Y] = t_1[Y]$  and  $t_2[Y, Y_p] \asymp t_p[Y, Y_p]$ . Intuitively, the CIND enforces the inclusion dependency  $R_1[X] \subseteq R_2[Y]$  for each tuple that matches the pattern for  $[X, X_p]$  and also requires that the tuple in  $R_2$  should match the pattern for  $[Y, Y_p]$ .

Instance in Fig. 7 does not satisfy constraint  $\psi_2$  since there is a journalist with *Num* = 05 in relation *Staff* for which there is no tuple in *Journalist* with *SNum* = 05. It also violates constraint  $\psi_3$  because for tuple (01,news) in relation *Journalist*, there is no tuple in *Staff* with *Num* = 01 and *Type* = “journalist”.

Classical functional dependencies are a particular case of CFDs for which the pattern tableau has a single tuple with only unnamed variables. In the same way, classical inclusion dependencies are a particular case of CINDs that can be defined using  $X_p = Y_p = \text{nil}$  and a pattern tableau with a single tuple with only unnamed variables.

CFDs have been extended to consider (a) disjunction and inequality [23], (b) ranges of values [57], and (c) cardinality and synonym rules [33]. Automatic

---

<sup>3</sup>The empty list is denoted by *nil*.

generation and discovery of CFDs have been studied in [34, 43, 46]. Pattern tableaux have also been used to show the portions of the data that satisfy (or violate) a constraint [57, 58]. These pattern tableaux can be used both to characterize the quality of the data and generate CFDs and CINDs.

We can see that conditional dependencies (CDs) have a classic semantics. Actually, it is easy to express them in first-order predicate logic. However, they have been proposed with the problems of data quality and data cleaning in mind.

## 4.2 Data Cleaning with CDs

CFDs and CINDs were introduced to specify data dependencies that were not captured by classic constraints and to be used for improving the quality of the data. In [37] a data cleaning framework is provided for conditional functional dependencies. Given a database instance which is inconsistent with respect to a set of CFDs, the main objective is to find a repair through attribute updates that minimizes a cost function. The cost of changing a value  $v$  by a value  $v'$  is  $\text{cost}(v, v') = w(t, A) \cdot \text{dist}(v, v') / \max(|v|, |v'|)$ , where  $w$  is a weight assigned to attribute  $A$  in tuple  $t$  and  $\text{dist}$  is a distance function. The weights, for example, could be used to include in the repair process information that we could have about the data, e.g., about their accuracy. The distance function should measure the similarity between values. It could be, for example, the *edit distance* that counts the number of insertions and deletions of characters to transform from one string into another.

Consider relation  $S$  in Fig. 8a, the conditional FDs  $\varphi_1 = (X \rightarrow Y, T_1)$ ,  $\varphi_2 = (Y \rightarrow W, T_2)$ , and  $\varphi_3 = (Z \rightarrow W, T_3)$ , with:

$T_1:$	$X$	$Y$
	-	-

$T_2:$	$Y$	$W$
	b	-

$T_3:$	$Z$	$W$
	c	d
	k	e

The instance does not satisfy  $\varphi_1$ . Solving the inconsistencies with respect to  $\varphi_1$  by replacing  $t_1[Y]$  by  $b$  triggers more inconsistencies with respect to  $\varphi_2$ , which need to be solved by changing more values for attributes  $Z$  and  $W$ . On the other hand, if we instead replace  $t_2[Y]$  and  $t_3[Y]$  by  $c$ , all inconsistencies are solved. The repairs obtained in these two ways correspond to  $S_1$  and  $S_2$  in Fig. 8 (the updated values are in bold). If we consider  $w(t, A) = 1$  for all attributes and use the edit distance, the cost of each change is 2, and therefore, the cost of repairs  $S_1$  and  $S_2$  are 6 and 4, respectively. Thus,  $S_2$  is a better repair than  $S_1$ . Another Minimal-cost repair, say  $S_3$ , is obtained by replacing  $t_2[Y]$  by  $c$ , and  $t_3[X]$  by any constant different from  $a$ . There are some situations where inconsistencies with respect to CFDs cannot be solved by replacing values by constants of the domain. In those cases, repairs are obtained using *null*. A tuple with *null* will not create a new inconsistency with respect to an existing constraint since we assume that  $t[X] \not\propto t_p[X]$  when  $t[X]$  contains *null*. For example, assume that we add the constraint  $(U \rightarrow W, T_4)$ , with

<b>a</b>	<b>b</b>	<b>c</b>
$X \mid Y \mid Z \mid W \mid U$	$X \mid Y \mid Z \mid W \mid U$	$X \mid Y \mid Z \mid W \mid U$
$t_1: a \ c \ c \ d \ f$	$t_1: a \ b \ c \ d \ f$	$t_1: a \ c \ c \ d \ f$
$t_2: a \ b \ k \ e \ g$	$t_2: a \ b \ c \ d \ g$	$t_2: a \ c \ k \ e \ g$
$t_3: a \ b \ c \ d \ h$	$t_3: a \ b \ c \ d \ h$	$t_3: a \ c \ c \ d \ h$
$S$	Repair $S_1$	Repair $S_2$

**Fig. 8** Inconsistent instance  $S$  and two possible repairs  $S_1$  and  $S_2$

$T_4 = \{(\_, d), (\_, e)\}$ , requiring that every tuple in  $S$  should contain  $d$  and  $e$  in attribute  $W$ . Enforcing it is, of course, not possible, unless we satisfy the constraint by using *null*.

The problem of finding a minimal-cost repair is *coNP*-complete, but an efficient approximation algorithm, based on equivalent classes, is provided in [37]. The repair process is guided by interaction with the user, to ensure its accuracy. As a way to minimize the required feedback, it is possible to add machine learning capabilities and to learn from previous choices by the user [75]. If the cost of each update depends only on the tuple that is being updated, i.e.,  $cost(v, v') = w(t)$ , it is possible to find a constant factor approximation of a repair when the set of constraints is fixed [61].

## 5 Applications of Declarative Approaches to Entity Resolution

The problem of *entity resolution* (ER) is about discovering and matching database records that represent the same entity in the application domain, i.e., detecting and solving *duplicates* [17, 41, 66]. ER is a classic, common, and difficult problem in data cleaning, for which several ad hoc and domain-dependent mechanisms have been proposed.

ER is a fundamental problem in the context of data analysis and decision making in business intelligence. From this perspective, it becomes particularly crucial in data integration [65] and even more difficult in virtual data integration systems (VDIS). As we saw in Sect. 3.3, logic-based specifications of the intended solutions of a generic VDIS have been proposed, used, and investigated [11, 62]. As a consequence, logic-based specifications of ER or generic approaches to ER, which could be combined with the specifications of the integration process, would be particularly relevant.

Notice that in virtual data integration systems, sources are usually not modified through the mediator. As a consequence, physical ER through the integration system is not possible. This forces us to consider as a real alternative some form of on-the-fly ER, performed at query answering time. The declarative, logic-based approaches to ER are particularly appropriate for their amalgamation with queries and query answering processes via some sort of query rewriting.

## 5.1 A Generic Approach: Swoosh

In [7], a generic conceptual framework for entity resolution is introduced, the *Swoosh* approach. It considers a general match relation  $M$  and a general merge function,  $\mu$ . In the main general formulation of Swoosh, the match relation  $M$  and the merge function  $\mu$  are defined at the *record* (or tuple) level (but see [7] for some extensions). That is, when two records in a database instance are matched (found to be similar), they can be merged into a new record. This is iteratively done until the *entity resolution* of the instance is computed. Due to the merge process, some database tuples (or records) may be discarded. More precisely, the number of tuples may decrease during the ER process, because tuples that are *dominated* by others are eliminated (see below).

Swoosh views a database instance  $I$  as a finite set of records  $I = \{r_1, \dots, r_n\}$  taken from an infinite domain of records  $Rec$ . Relation  $M$  maps  $Rec \times Rec$  into  $\{\text{true}, \text{false}\}$ . When two records are similar (and then could be merged),  $M$  takes the value *true*. Moreover,  $\mu$  is a partial function from  $Rec \times Rec$  into  $Rec$ . It produces the merge of two records into a new record and is defined only when  $M$  takes the value *true*.

Given an instance  $I$ , the *merge closure* of  $I$  is defined as the smallest set of records  $\bar{I}$ , such that  $I \subseteq \bar{I}$ , and for every two records  $r_1, r_2$  for which  $M(r_1, r_2) = \text{true}$ , it holds  $\mu(r_1, r_2) \in \bar{I}$ . The merge closure of an instance is unique and can be obtained by adding merges of matching records until a fixpoint is reached.

Swoosh considers a general domination relationship between two records  $r_1, r_2$ , written as  $r_1 \leq_s r_2$ , which means that the information in  $r_1$  is subsumed by the information in  $r_2$ . Going one step further, we say that instance  $I_2$  dominates instance  $I_1$ , denoted  $I_1 \sqsubseteq_s I_2$ , whenever every record of  $I_1$  is dominated by some record in  $I_2$ .

For an instance  $I$ , an *entity resolution* is defined as a subset-minimal set of records  $I'$ , such that  $I' \subseteq \bar{I}$  and  $\bar{I} \sqsubseteq_s I'$ . It is shown that for every instance  $I$ , there is a unique entity resolution  $I'$  [7], which can be obtained from the merge closure by removing records that are dominated by other records.

A particularly interesting case of Swoosh occurs when the match relation  $M$  is reflexive and symmetric, and the merge function  $\mu$  is idempotent, commutative, and associative. We then use the domination order imposed by the merge function, which is defined by  $r_1 \leq_s r_2$  if and only if  $\mu(r_1, r_2) = r_2$ . Under these assumptions, the merge closure and therefore the entity resolution of every instance are finite [7].<sup>4</sup>

We can see that Swoosh's framework is generic and abstract enough to accommodate different forms of ER in different domains. Now, a still generic but special case of Swoosh, which also captures common forms of ER, is the so-called *union case*, that does matching, merging, and merge domination at the attribute level [7]. We illustrate this case by means of an example.

---

<sup>4</sup>Finiteness is shown for the case when match and merge have the *representativity* property (equivalent to being similarity preserving) in addition to other properties. However, the proof in [7] can be modified so that representativity is not necessary.

*Example 1.* We can treat records as *objects*, i.e., as sets of attribute/value pairs. In this case, a common way of merging records is via their union, as objects. For example, consider:

$$\begin{aligned} r_1 &= \{\langle Name, \{J. Doe\} \rangle, \langle St.Number, \{55\} \rangle, \langle City, \{Toronto\} \rangle\}, \\ r_2 &= \{\langle Name, \{J. Doe\} \rangle, \langle Street, \{Grenadier\} \rangle, \langle City, \{Vancouver\} \rangle\}. \end{aligned}$$

If they are considered to be similar, e.g., on the basis of their values for attribute *Name*, they can be merged into:

$$\mu(r_1, r_2) = \{\langle Name, \{J. Doe\} \rangle, \langle St.Number, \{55\} \rangle, \langle Street, \{Grenadier\} \rangle, \langle City, \{Toronto, Vancouver\} \rangle\}. \quad \square$$

In the union-case, one obtains a single resolved instance (i.e., a single entity resolution).

Swoosh has been extended in [73] with *negative rules*. They are used to avoid inconsistencies (e.g., with respect to semantic constraints) that could be introduced by indiscriminate matching. From this point of view, certain elements of *database repairs* (cf. Sect. 3) are introduced into the picture (cf. [73, Sect. 2.4]). In this direction, the combination of database repairing and ER is studied in [47].

## 5.2 ER with Matching Dependencies

Matching dependencies (MDs) are declarative rules that generalize entity resolution (ER) in relational DBs [44, 45]. They specify attribute values that have to be made equal under certain conditions of similarity for other attribute values.

*Example 2.* Consider the relational schema  $R_1(X, Y), R_2(X, Y)$ , where  $X$  and  $Y$  are attributes (or lists of them). The following symbolic expression is a matching dependency:

$$\varphi : R_1[\bar{X}_1] \approx R_2[\bar{X}_2] \longrightarrow R_1[A_1] \doteq R_2[A_2]. \quad (5)$$

It says that “when in two tuples the values for attribute(s)  $\bar{X}$  in  $R_1, R_2$  are similar, the values in them for attribute(s)  $A$  must be matched/merged, i.e., made equal.”  $\square$

In an MD like this  $R_1$  and  $R_2$  can be the same predicate if we are partially merging tuples of a same relation. The similarity relation,  $\approx$ , is application-dependent, associated to a particular attribute domain. It is assumed to be reflexive and symmetric.<sup>5</sup>

---

<sup>5</sup>Notice that the similarity relation in this section corresponds somehow to the match function  $M$  of Sect. 5.1; and matching functions we will consider here to identify two values, to the merge function  $\mu$  of Sect. 5.1.

*Example 3.* Now consider the MD  $\varphi$  telling us that “*similar name and phone number  $\rightarrow$  identical address*”. We apply it to the initial instance  $D_0$  as follows:

$D_0$	<i>Name</i>	<i>Phone</i>	<i>Address</i>	
	John Doe	(613)123 4567	Main St., Ottawa	
	J. Doe	123 4567	<u>25 Main St.</u>	
				$\implies$
$D_1$	<i>Name</i>	<i>Phone</i>	<i>Address</i>	
	John Doe	(613)123 4567	25 Main St., Ottawa	
	J. Doe	123 4567	25 Main St., Ottawa	

We can see that, in contrast to classical and conditional dependencies, which have a “static semantics,” an MD has a *dynamic semantics* that requires a pair of databases:  $(D_0, D_1) \models \varphi$ . That is, when the left-hand side (LHS) of  $\varphi$  is satisfied in  $D_0$ , the RHS (the matching) is made true in a second instance  $D_1$ .

In this example we are using a very particular matching function (MF) that implicitly treats attribute values as objects, i.e., sets of pairs  $\langle \text{Attribute}, \text{Value} \rangle$ , e.g., the first value for *address* in  $D_0$  can be seen as  $\{\langle \text{StreetName}, \text{MainSt.} \rangle, \langle \text{City}, \text{Ottawa} \rangle, \langle \text{HouseNumber}, \epsilon \rangle\}$ . The MF produces the *union* of the two records as sets and next, for a same attribute, also the union of local values.  $\square$

An MD does not tell us how to do the matching. In this regard, two alternatives have been explored in the literature. One of them treats the values in common required by the matching on the RHS essentially as an existential quantification. For example, for (5), there is a value  $y$  for which  $R_1[Y_1] = y = R_2[Y_2]$ . The initial instance can be “chased” with the MDs, producing a duplicate free instance (as prescribed by the set of MDs). Desirable clean instances could be those that minimize the number of changes of attribute values. We refer to [53–55] for details on this approach.

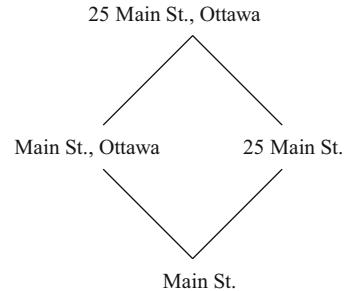
The second approach uses matching functions to provide a value in common for a matching. We briefly describe it in the following (details can be found in [13, 15]).

Given the MD (5) over a database schema, we say that the pair of instances  $(D, D')$  satisfies  $\varphi$ , denoted  $(D, D') \models \varphi$ , iff every pair of tuples for which  $D$  satisfies the antecedent but not the consequent of  $\varphi$ , the consequent is made true (satisfied) in  $D'$  (by matching attribute values in those tuples as prescribed by function  $m$ ). Formally,  $(D, D') \models \varphi$  when for every  $R_1$ -tuple  $t_1$  and  $R_2$ -tuple  $t_2$ : If  $t_1[\bar{X}] \approx t_2[\bar{X}]$ , but  $t_1[A_1] = a_1 \neq t_2[A_2] = a_2$  in  $D$ , then  $t_1[A_1] = t_2[A_2] = m_A(a_1, a_2)$  in  $D'$ .

Above,  $m_A$  is a binary idempotent, commutative and associative function defined on the domain in common of attributes  $A_1$  and  $A_2$ . The MF  $m_A$  induces a finite semi-lattice with partial order defined by  $a \leq_A a' : \Leftrightarrow m_A(a, a') = a'$ . That is,  $a'$  dominates  $a$  when  $a$  and  $a'$  are matched into  $a'$ . Furthermore, the least-upper-bound of two elements in the lattice is obtained by applying function  $m$  to them:  $\text{lub}_{\leq} \{a, a'\} = m_A(a, a')$ . We usually assume the greatest-lower-bound of any two lattice elements,  $\text{glb}_{\leq} \{a, a'\}$ , also exists.

MFs can be used to define a semantic-domination lattice, related to information contents, as in “domain theory” [25]. The higher in the lattice, the more information contents. Figure 9 shows an information lattice at the domain level.

**Fig. 9** A semantic-domination lattice



Now, for a set  $\Sigma$  of MDs, an instance  $D'$  is *stable* if  $(D', D') \models \Sigma$ . The idea is to obtain a stable instance by applying the MDs starting from an initial instance  $D$ . This defines a *chase procedure* by which different chase sequences can be generated:

$D \xrightarrow{\varphi_1} D_1 \xrightarrow{\varphi_2} D_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_n} D'$   
 dirty instance stable (clean) instance

The final (finitely terminating) results of the chase are the so-called *clean instances*. They form a class that we denote with  $\text{Clean}(D, \Sigma)$ . Since tuples identifiers are used, it is possible to put in correspondence a clean instance with the original instance.

The partial orders  $\preceq_A$  at the attribute level can be lifted to a partial order on tuples of a same relation:  $t_1 \preceq t_2 : \iff t_1[A] \preceq_A t_2[A]$ , for all  $A$ , which in its turn gives rise to a partial order of *semantic domination* at the instance level (of a same relation):  $D_1 \sqsubseteq D_2 : \iff \forall t_1 \in D_1 \exists t_2 \in D_2 \ t_1 \preceq t_2$ .

When constructing the partial order  $\sqsubseteq$ , we get rid of dominated tuples within a relation. This partial order is useful to compare sets of query answers as instances of a relation. Actually, we use it to define the set,  $\text{CleanAns}_D^\Sigma(\mathcal{Q})$ , of *clean answers* to a query  $\mathcal{Q}$  posed to a (possibly dirty) instance  $D$  that is subject to a set  $\Sigma$  of MDs. Intuitively, they are the answers that are invariant under the ER process:  $\text{CleanAns}_D^\Sigma(\mathcal{Q}) := \text{glb}_{\sqsubseteq} \{\mathcal{Q}(D') \mid D' \in \text{Clean}(D, \Sigma)\}$ . Notice that this is a lattice-dependent notion of “certain” answer.

*Example 4.* Consider the MD  $\varphi : R[name] \approx R[name] \rightarrow R[address] \doteq R[address]$  applied to instance  $D_0$  below, whose clean instances are  $D'$  and  $D''$ .

$D_0$	$Name$	$Address$
	John Doe	Main St., Ottawa
	J. Doe	25 Main St.
	Jane Doe	25 Main St., Vancouver

$D'$	$Name$	$Address$	$D''$	$Name$	$Address$
	John Doe	25 Main St., Ottawa		John Doe	Main St., Ottawa
	J. Doe	25 Main St., Ottawa		J. Doe	25 Main St., Vancouver
	Jane Doe	25 Main St., Vancouver		Jane Doe	25 Main St., Vancouver

For the query  $\mathcal{Q}$  :  $\pi_{address}(\sigma_{name=\text{"J. Doe"}}(R))$ , it holds:  $ClearAns_D^\Sigma(\mathcal{Q}) = \{25\text{ Main St.}\}$ .  $\square$

It is possible to prove that every chase sequence terminates in polynomially many steps in the size of the original instance. The result is a clean instance (by definition) that semantically dominates the original instance. Furthermore, computing clean answers is a *coNP*-complete problem (in data complexity) [15].

### 5.3 Answer-Set Programs for MD-Based ER

A natural research goal is to come up with a general methodology to specify the result of an MD-based ER process. More precisely, the aim is to compactly and declaratively specify the class of clean instances for an instance  $D$  subject to ER on the basis of a set  $\Sigma$  of MDs. In principle, a logic-based specification of that kind could be used to reason about/from the class of clean instances, in particular, enabling a process of clean query answering.

A simple but important observation about MD-based ER (or ER in general) is that clean query answering becomes a non-monotonic process, in the sense that we can lose clean answers when the database is updated and has to undergo a new ER process. As a consequence, the specification mentioned above must appeal to some sort of non-monotonic logical formalism. Actually, it is possible to use (disjunctive) *answer set programs* (ASPs), in such a way that the class of stable models of the “cleaning program,” say  $\Pi(D, \Sigma)$ , corresponds to the class  $Clean(D, \Sigma)$  of clean instances. On this basis, the clean answers to a query posed to  $D$  can be obtained via *cautious reasoning* from the program. In the following, we illustrate the approach by means of an extended example; details can be found in [4].

The main idea is that program  $\Pi(D, \Sigma)$  implicitly simulates the chase sequences, each one represented by a model of the program. For this,  $\Pi(D, \Sigma)$  has rules to (1) enforce MDs on pairs of tuples satisfying similarities conditions, (2) create newer versions of those tuples by applying MFs, (c) make the older versions of the tuples unavailable for other matchings, and (d) make each stable model correspond to a *valid chase sequence*, leading to a clean instance. This is the most intricate part.

In order to give an example of cleaning program, consider the set  $\Sigma$  of MDs, matching function  $M_B$ , similarity relation, and initial instance below:

$$\varphi_1: R[A] \approx R[A] \rightarrow R[B] \doteq R[B], \quad \varphi_2: R[B] \approx R[B] \rightarrow R[B] \doteq R[B].$$

		$R(D)$	$A$	$B$
$M_B(b_1, b_2, b_{12})$	$a_1 \approx a_2$	$t_1$	$a_1$	$b_1$
$M_B(b_2, b_3, b_{23})$	$b_1 \approx b_2$	$t_2$	$a_2$	$b_2$
$M_B(b_1, b_{23}, b_{123})$		$t_3$	$a_3$	$b_3$

Enforcing  $\Sigma$  on  $D$  results in two chase sequences:

$D$	$A$	$B$	$D_1$	$A$	$B$	$D$	$A$	$B$	$D'_1$	$A$	$B$
$t_1$	$a_1$	$b_1$	$t_1$	$a_1$	$b_{12}$	$t_1$	$a_1$	$b_1$	$t_1$	$a_1$	$b_1$
$t_2$	$a_2$	$b_2$	$t_2$	$a_2$	$b_{12}$	$t_2$	$a_2$	$b_2$	$t_2$	$a_2$	$b_{23}$
$t_3$	$a_3$	$b_3$	$t_3$	$a_3$	$b_3$	$t_3$	$a_3$	$b_3$	$t_3$	$a_3$	$b_{23}$
									$D'_2$	$A$	$B$
									$t_1$	$a_1$	$b_{123}$
									$t_2$	$a_2$	$b_{123}$
									$t_3$	$a_3$	$b_{23}$
			$\Rightarrow_{\varphi_1}$								

Program  $\Pi(D, \Sigma)$  contains:

1. For every tuple (id)  $t^D = R(\bar{c})$ , the fact  $R'(t, \bar{c})$ , i.e., we use explicit tuple IDs.
2. Rules to capture possible matchings when similarities hold for two tuples:

$$\begin{aligned} Match_{\varphi_1}(T_1, X_1, Y_1, T_2, X_2, Y_2) \vee NotMatch_{\varphi_1}(T_1, X_1, Y_1, T_2, X_2, Y_2) \leftarrow \\ R'(T_1, X_1, Y_1), R'(T_2, X_2, Y_2), X_1 \approx X_2, Y_1 \neq Y_2. \end{aligned} \quad (\text{similarly for } Match_{\varphi_2})$$

Here,  $T_i$  stands for a tuple ID. Notice that predicate  $Match$  does not do the actual merging, and we need the freedom to match or not to match, to obtain different chase sequences (cf. below).

3.  $Match$  does not take place if one of the involved tuples is used for another matching and replaced by newer version:

$$\begin{aligned} \leftarrow NotMatch_{\varphi_1}(T_2, X_2, Y_2, T_1, X_1, Y_1), not OldVersion(T_1, X_1, Y_1), \\ not OldVersion(T_2, X_2, Y_2). \end{aligned}$$

This is a *program constraint* filtering out models that make the body true (similarly for  $NotMatch_{\varphi_2}$ ).

4. Predicate  $OldVersion$  is specified as containing different versions of every tuple in relation  $R'$  which has been replaced by a newer version. This is captured by *upward lattice navigation*:

$$OldVersion(T_1, \bar{Z}_1) \leftarrow R'(T_1, \bar{Z}_1), R'(T_1, \bar{Z}'_1), \bar{Z}_1 \preceq \bar{Z}'_1, \bar{Z}_1 \neq \bar{Z}'_1.$$

Up to this point we have no rules to do the actual merging yet.

5. Rules to insert new tuples by merging, creating new versions:

$$\begin{aligned} R'(T_1, X_1, Y_3) &\leftarrow Match_{\varphi_1}(T_1, X_1, Y_1, T_2, X_2, Y_2), M_B(Y_1, Y_2, Y_3). \\ R'(T_1, X_1, Y_3) &\leftarrow Match_{\varphi_2}(T_1, X_1, Y_1, T_2, X_2, Y_2), M_B(Y_1, Y_2, Y_3). \end{aligned}$$

The rules so far tell us what can be done or not in terms of matching and merging, but not exactly how to combine those possibilities. So, we need additional structure to create valid chase sequences. Since each chase sequence is an ordered sequence of instances within a partial order of instances, these ordered sequences have to be captured by additional conditions, which we do next.

6. Rules specifying a predicate  $Prec$  that records the relative order of matchings. It applies to two pairs of tuples, to two matchings of two tuples. For MDs  $\varphi_j, \varphi_k \in \{\varphi_1, \varphi_2\}$ :

$$\begin{aligned} Prec(T_1, X_1, Y_1, T_2, X_2, Y_2, T_1, X_1, Y'_1, T_3, X_3, Y_3) &\leftarrow Match_{\varphi_j}(T_1, X_1, Y_1, T_2, X_2, Y_2), \\ &Match_{\varphi_k}(T_1, X_1, Y'_1, T_3, X_3, Y_3), Y_1 \preceq Y'_1, Y_1 \neq Y'_1. \end{aligned}$$

This predicate is better read as  $Prec(\langle T_1, X_1, Y_1 \rangle, \langle T_2, X_2, Y_2 \rangle \mid \langle T_1, X_1, Y_1 \rangle, \langle T_3, X_3, Y_3 \rangle)$ , saying that the matching of the first two tuples precedes the matching of the last two. For two matchings applicable to two different versions of a tuple,  $Prec$  records their relative order, with matching applied to  $\preceq$ -smaller version first.

A couple of similar rules are still required (see [4] for more details). With this definition,  $Prec$  could still not be an order relation, e.g., it could violate antisymmetry. Consequently, program constraints are used to make it an order, and each stable model will have a particular version of that order. That is, different orders correspond to different models and to different chase sequences. More precisely, additional rules and program constraints are needed to make  $Prec$  reflexive, antisymmetric, and transitive (not given here). They are used to eliminate instances (models) that result from illegal applications of MDs.

In essence, what we have done here is to define a predicate  $Prec$  in terms of the matching of tuples. By imposing restrictions on this predicate, we implicitly impose conditions on the matchings that, in their turn, are the basis for the actual merging, i.e., applications of the matching functions.

Finally, rules are introduced to collect the latest version of each tuple, to form the clean instance, with new predicates (nicknames):

$$R^c(T_1, X_1, Y_1) \leftarrow R'(T_1, X_1, Y_1), \text{not } OldVersion(T_1, X_1, Y_1).$$

This example illustrates a general and computable methodology to produce cleaning ASPs from an instance  $D$  and a set  $\Sigma$  of MDs. With it we obtain a single logical specification of all chase sequences. It is possible to prove that there is a one-to-one correspondence between  $Clean(D, \Sigma)$  and the stable models of  $\Pi(D, \Sigma)$ . More precisely, the clean instances are the restrictions to predicates  $R^c$  of the stable models.

We can use the same program  $\Pi(D, \Sigma)$  to compute clean answers to different queries  $\mathcal{Q}$  posed to  $D$ . The query has to be first expressed in terms of the  $R^c$ -atoms. For example, the query  $\mathcal{Q}(x) : \exists x' y (R(x, y) \wedge R(x', y) \wedge x \neq x')$  becomes the rule (a simple query program)  $Ans_{\mathcal{Q}}(X) \leftarrow R^c(T, X, Y), R^c(T', X', Y), X \neq X'$ , which has to be added to the cleaning program. Clean answers can be obtained by set-theoretic cautious reasoning from the combined program, i.e., as the intersection of the sets of answers in all stable models [4].

As we can see, the cleaning programs provide a general methodology for clean query answering. However, a natural question is whether these programs are too expressive for this task. In this regard, it is possible to verify that the syntactic structure of the cleaning programs makes them *head-cycle free*. As a consequence, they can be transformed into equivalent non-disjunctive programs, for which cautious query answering is *coNP*-complete in data [38]. This matches the intrinsic complexity of query answering (cf. Sect. 5.2).

As a final remark, we make notice that the clean answers were not defined via a set-theoretic intersection (as usual in ASP), but via the lattice-theoretic *glb* (cf. Sect. 5.2). The clean answers can still be computed via ASPs by introducing some additional rules into the query program that capture the *glb* in set-theoretic terms [4].

## 5.4 MDs and Swoosh

The Swoosh's ER methodology is generic, but not declarative, in the sense that the semantics of the system is not captured in terms of a logical specification of the instances resulting from the cleaning process. On the other side, MD-based ER is initially based on a set of matching dependencies, and the semantics is model-theoretic, as captured by the clean instances. However, the latter have a procedural component. A really declarative specification of MD-based ER is eventually given through the cleaning programs introduced in the previous section.

In [7], algorithms are introduced for different cases of Swoosh. One of them, instead of working at the full record level (cf. Sect. 5.1), considers doing the matching on the basis of values for *features*, which consider certain combinations of attributes [7, Sect. 4]. This is in some sense close to the spirit of MDs.

In [13, 15], Swoosh's union-case is reconstructed via MDs. This indicates that it is possible to apply the general methodology for writing cleaning programs presented in Sect. 3.1 for that case of Swoosh. Here we show instead a direct methodology for producing this kind of cleaning programs. In this way, we obtain a declarative and executable version of the Swoosh's union-case.

Actually, for each instance of this case of Swoosh, it is possible to construct a non-disjunctive stratified ASP,  $\Pi^{UC}(D)$ , that uses function and set terms [29], for set union and set membership. Such a program has a single stable model that corresponds to the unique resolved instance guaranteed by Swoosh, and it can be computed in polynomial time in data. Here we only give an example:

*Example 5.* Assume the matchings  $a_1 \approx_{\underline{A}} a_2$ ,  $a_2 \approx_{\underline{A}} a_3$  hold. Records have attributes  $A$  and  $B$ , whose values are sets of elements of the underlying domains  $\underline{A}$  and  $\underline{B}$ , respectively. Here, two records matching in  $A$  are fully merged, and two set values match if there are  $\underline{A}$ -elements in them that match. The following is a resolution process based on the union-case:

$R(D)$	$R(D')$
$\begin{array}{ c c } \hline A & B \\ \hline \{a_1\} & \{b_1\} \\ \{a_2\} & \{b_2\} \\ \{a_3\} & \{b_3\} \\ \hline \end{array}$	$\Rightarrow$
	$\begin{array}{ c c } \hline A & B \\ \hline \{a_1, a_2\} & \{b_1, b_2\} \\ \{a_2, a_3\} & \{b_2, b_3\} \\ \hline \end{array}$
	$\Rightarrow$
	$\begin{array}{ c c } \hline A & B \\ \hline \{a_1, a_2, a_3\} & \{b_1, b_2, b_3\} \\ \hline \end{array}$

The cleaning program  $\Pi^{UC}(D)$  contains (1) facts for the initial instance  $D$ , plus  $match_{\underline{A}}(a_1, a_2)$ ,  $a_1, a_2 \in Dom_{\underline{A}}$ , and (2) rules for the merge closure of  $D$ :

$$\begin{aligned} R(\#Union(S_1^1, S_1^2), \#Union(S_2^1, S_2^2)) &\leftarrow R(S_1^1, S_2^1), R(S_1^2, S_2^2), \\ &\quad \#Member(A_1, S_1^1), \#Member(A_2, S_2^2), \\ &\quad Match_{\underline{A}}(A_1, A_2), (S_1^1, S_2^1) \neq (S_1^2, S_2^2). \end{aligned}$$

Tuple domination is captured via subset relation:

$$\begin{aligned} Dominated_R(S_1^1, S_2^1) &\leftarrow R(S_1^1, S_2^1), R(S_1^2, S_2^2), (S_1^1, S_2^1) \neq (S_1^2, S_2^2) \\ (\#Union(S_1^1, S_1^2), \#Union(S_2^1, S_2^2)) &= (S_1^2, S_2^2). \end{aligned}$$

Finally, the elements of the ER are collected:

$$R^{Er}(S_1, S_2) \leftarrow R(S_1, S_2), \text{ not Dominated}_R(S_1, S_2).$$

□

## 5.5 Rules and Ontologies for Duplicate Detection

In the previous sections we have mostly concentrated on the merge part of ER. However, identifying similarities and duplicates is also an important and common problem [66]. There are some declarative approaches to duplicate detection. They could be naturally combined with declarative approaches to merging.

A declarative framework for collective entity matching of large data sets using domain-specific soft and hard constraints is proposed in [1]. The constraints specify the matchings. They use a novel Datalog style language, *Dedupalog*, to write the constraints as rules. The matching process tries to satisfy all the hard constraints but minimizing the number of violations to the soft constraints. Dedupalog is used for identifying groups of tuples that could be merged. They do not do the merging or base their work on MDs.

Another declarative approach to ER is presented in [69]. The emphasis is placed mainly on the detection of duplicates rather than on the actual merging. An ontology expressed in a logical language based on RDF-S, OWL-DL, and SWRL is used for this task. Reconciliation rules are captured by SWRL, a rule language for the semantic web. Also negative rules that prevent reconciliation of certain values can be expressed, much in the spirit of Swoosh with negative rules [73].

## 6 Final Remarks

In the previous sections we have presented some recent developments in the area of declarative and generic data cleaning. The proposed solutions are general enough to be applied in different cases. Furthermore, the semantics of those approaches are formal and precise and also executable on the basis of their logical and symbolic formulations.

In Sect. 5 we have restricted ourselves to some recent generic and declarative approaches to entity resolution, which is vast subject, with a solid body of research. For more details and a broader perspective of entity resolution, we refer the reader to [70].

We have left out of this discussion several problems and approaches in data cleaning that can also be approached from a declarative and generic side. One of them is *data editing* [49], which is particularly crucial in census data and can also treated with logic-based methods [20, 51]. Master data [8] are used as a reference for different data cleaning tasks, e.g., entity resolution and data editing. The use of master data in combination with data editing and database repairing has been recently investigated in [48].

We are just starting to see the inception of generic and declarative approaches to data quality assessment and data cleaning. An interesting direction to watch is the one of *ontology-based data management* [63], which should naturally lead to *ontology-based data quality*. Ontologies provide both the semantics and the contexts upon which the activities of data quality assessment and data cleaning naturally rely [14, 64].

**Acknowledgements** This chapter describes research supported by the NSERC Strategic Network on Business Intelligence (BIN), NSERC/IBM CRDPJ/371084-2008, NSERC Discovery, and Bicentenario Project PSD-57. We are grateful to our research collaborators with whom part of the research described here has been carried out.

## References

1. Arasu A, Ré C, Suciu D (2009) Large-scale deduplication with constraints using dedupalog. In: Proceedings of the 2009 IEEE international conference on data engineering (ICDE'09). IEEE Computer Society
2. Arenas M, Bertossi L, Chomicki J (1999) Consistent query answers in inconsistent databases. In: Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems (PODS'99). ACM, New York, pp 68–79
3. Arenas M, Bertossi L, Chomicki J (2003) Answer sets for consistent query answering in inconsistent databases. *Theory Pract Logic Program* 3(4):393–424
4. Bahmani Z, Bertossi L, Kolahi S, Lakshmanan LVS (2012) Declarative entity resolution via matching dependencies and answer set programs. In: Brewka G, Eiter T, McIlraith SA (eds) Proceedings of the 13th international conference on principles of knowledge representation and reasoning (KR'12). AAAI Press, pp 380–390
5. Barceló P, Bertossi L, Bravo L (2001) Characterizing and computing semantically correct answers from databases with annotated logic and answer sets. In: Bertossi L, Katona GOH, Schewe KD, Thalheim B (eds) Semantics in databases, LNCS, vol. 2582. Springer, Berlin, pp 7–33
6. Batini C, Scannapieco M (2006) Data quality: concepts, methodologies and techniques. Data-centric systems and applications. Springer, Berlin
7. Benjelloun O, Garcia-Molina H, Menestrina D, Su Q, Whang SE, Widom J (2009) Swoosh: a generic approach to entity resolution. *VLDB J* 18(1):255–276
8. Berson A, Dubov L (2010) Master data management and data governance. McGraw-Hill Osborne Media
9. Bertossi L (2006) Consistent query answering in databases. *ACM SIGMOD Rec* 35(2):68–76
10. Bertossi L (2011) Database repairing and consistent query answering. Synthesis lectures on data management. Morgan & Claypool Publishers, San Rafael
11. Bertossi L, Bravo L (2005) Consistent query answers in virtual data integration systems. In: Bertossi L, Hunter A, Schaub T (eds) Inconsistency tolerance, LNCS, vol. 3300. Springer, Berlin, pp 42–83
12. Bertossi L, Bravo L, Franconi E, Lopatenko A (2008) The complexity and approximation of fixing numerical attributes in databases under integrity constraints. *Inform Syst* 33(4–5):407–434
13. Bertossi L, Kolahi S, Lakshmanan LVS (2011) Data cleaning and query answering with matching dependencies and matching functions. In: Proceedings of the 14th international conference on database theory (ICDT'11). ACM, New York, pp 268–279

14. Bertossi L, Rizzolo F, Jiang L (2011) Data quality is context dependent. In: Castellanos M, Dayal U, Markl V (eds) Proceedings of the 4th international workshop on enabling real-time business intelligence (BIRTE 2010) held at VLDB 2010, LNBI, vol. 84. Springer, Berlin, pp 52–67
15. Bertossi L, Kolahi S, Lakshmanan L (2012) Data cleaning and query answering with matching dependencies and matching functions. *Theory Comput Syst* 1–42. DOI 10.1007/s00224-012-9402-7
16. Blakeley JA, Coburn N, Larson PA (1989) Updating derived relations: detecting irrelevant and autonomously computable updates. *ACM Trans Database Syst* 14(3):369
17. Bleiholder J, Naumann F (2008) Data fusion. *ACM Comput Surv* 41(1):1–41
18. Bohannon P, Flaster M, Fan W, Rastogi R (2005) A cost-based model and effective heuristic for repairing constraints by value modification. In: Özcan F (ed) Proceedings of the ACM SIGMOD international conference on management of data. ACM, New York, pp 143–154
19. Bohannon P, Fan W, Geerts F, Jia X, Kementsietsidis A (2007) Conditional functional dependencies for data cleaning. In: Chirkova R, Dogac A, Özsu MT, Sellis TK (eds) Proceedings of the international conference on data engineering (ICDE 2007). IEEE
20. Boskovitz A, Goré R, Hegland M (2003) A logical formalisation of the fellegi-holt method of data cleaning. In: Berthold MR, Lenz HJ, Bradley E, Kruse R, Borgelt C (eds) Proceedings of the 5th international symposium on intelligent data analysis (IDA 2003), LNCS, vol. 2810. Springer, Berlin, pp 554–565
21. Bravo L, Bertossi L (2006) Semantically correct query answers in the presence of null values. In: Proceedings of the EDBT WS on inconsistency and incompleteness in databases (EDBT'06). Springer, Berlin, pp 336–357
22. Bravo L, Fan W, Ma S (2007) Extending dependencies with conditions. In: Koch C, Gehrke J, Garofalakis MN, Srivastava D, Aberer K, Deshpande A, Florescu D, Chan CY, Ganti V, Kanne CC, Klas W, Neuhold EJ (eds) Proceedings of the 33rd international conference on very large data bases (VLDB 2007). ACM, New York, pp 243–254
23. Bravo L, Fan W, Geerts F, Ma S (2008) Increasing the expressivity of conditional functional dependencies without extra complexity. In: Alonso G, Blakeley JA, Chen ALP (eds) Proceedings of the 24th international conference on data engineering (ICDE 2008). IEEE
24. Brewka G, Eiter T, Truszczynski M (2011) Answer set programming at a glance. *Commun ACM* 54(12):92–103
25. Buneman P, Jung A, Ohori A (1991) Using power domains to generalize relational databases. *Theor Comput Sci* 91(1):23–55
26. Calì A, Lembo D, Rosati R (2003) On the decidability and complexity of query answering over inconsistent and incomplete databases. In: Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems (PODS'03), pp 260–271
27. Calì A, Calvanese D, De Giacomo G, Lenzerini M (2004) Data integration under integrity constraints. *Inform Syst* 29:147–163
28. Calì A, Gottlob G, Lukasiewicz T, Marnette B, Pieris A (2010) Datalog+/-: a family of logical knowledge representation and query languages for new applications. In: Proceedings of the 25th annual IEEE symposium on logic in computer science (LICS 2010). IEEE Computer Society
29. Calimeri F, Cozza S, Ianni G, Leone N (2009) An ASP system with functions, lists, and sets. In: Erdem E, Lin F, Schaub T (eds) Proceedings of the 10th international conference on logic programming and nonmonotonic reasoning (LPNMR 2009), LNCS, vol. 5753. Springer, Berlin, pp 483–489
30. Caniupan M, Bertossi L (2010) The consistency extractor system: answer set programs for consistent query answering in databases. *Data Knowl Eng* 69(6):545–572
31. Caroprese L, Greco S, Zumpano E (2009) Active integrity constraints for database consistency maintenance. *IEEE Trans Knowl Data Eng* 21(7):1042–1058
32. Ceri S, Cochrane R, Widom J (2000) Practical applications of triggers and constraints: success and lingering issues (10-year award). In: El Abbadi A, Brodie ML, Chakravarthy S, Dayal U, Kamel N, Schlageter G, Whang KY (eds) Proceedings of the 26th international conference on very large data bases (VLDB 2000). Morgan Kaufmann, Orlando, pp 254–262

33. Chen W, Fan W, Ma S (2009) Incorporating cardinality constraints and synonym rules into conditional functional dependencies. *Inform Process Lett* 109(14):783–789
34. Chiang F, Miller RJ (2008) Discovering data quality rules. *VLDB J* 1(1):1166–1177
35. Chomicki J (2007) Consistent query answering: five easy pieces. In: Schwentick T, Suciu D (eds) Proceedings of the 11th international conference of database theory (ICDT 2007), LNCS, vol. 4353. Springer, Berlin, pp 1–17
36. Codd EF (1970) A relational model of data for large shared data banks. *Commun ACM* 13(6):377–387
37. Cong G, Fan W, Geerts F, Jia X, Ma S (2007) Improving data quality: consistency and accuracy. In: Koch C, Gehrke J, Garofalakis MN, Srivastava D, Aberer K, Deshpande A, Florescu D, Chan CY, Ganti V, Kanne CC, Klas W, Neuhold EJ (eds) Proceedings of the 33rd international conference on very large data bases (VLDB 2007). ACM, New York, pp 315–326
38. Dantsin E, Eiter T, Gottlob G, Voronkov A (2001) Complexity and expressive power of logic programming. *ACM Comput Surv* 33(3):374–425
39. Dong XL, Tan WC (2011) Letter from the special issue editors. *IEEE Data Eng Bull* 34(3):2
40. Eiter T, Fink M, Greco G, Lembo D (2008) Repair localization for query answering from inconsistent databases. *ACM Trans Database Syst* 33(2):10:1–10:51
41. Elmagarmid AK, Ipeirotis PG, Verykios VS (2007) Duplicate record detection: a survey. *IEEE Trans Knowl Data Eng* 19(1):1–16
42. Fan W (2008) Dependencies revisited for improving data quality. In: Lenzerini M, Lembo D (eds) Proceedings of the 27th ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems (PODS 2008). ACM, New York, pp 159–170
43. Fan W, Geerts F, Lakshmanan LVS, Xiong M (2009) Discovering conditional functional dependencies. In: Ioannidis YE, Lee DL, Ng RT (eds) Proceedings of the 25th international conference on data engineering (ICDE 2009). IEEE
44. Fan W, Jia X, Li J, Ma S (2009) Reasoning about record matching rules. *VLDB J* 2(1):407–418
45. Fan W, Gao H, Jia X, Li J, Ma S (2011) Dynamic constraints for record matching. *VLDB J* 20(4):495–520
46. Fan W, Geerts F, Li J, Xiong M (2011) Discovering conditional functional dependencies. *IEEE Trans Knowl Data Eng* 23(5):683–698
47. Fan W, Li J, Ma S, Tang N, Yu W (2011) Interaction between record matching and data repairing. In: Sellis TK, Miller RJ, Kementsietsidis A, Velegrakis Y (eds) Proceedings of the ACM SIGMOD international conference on management of data. ACM, New York, pp 469–480
48. Fan W, Li J, Ma S, Tang N, Yu W (2012) Towards certain fixes with editing rules and master data. *VLDB J* 21(2):213–238
49. Fellegi IP, Holt D (1976) A systematic approach to automatic edit and imputation. *J Am Stat Assoc* 71(353):17–35
50. Flesca S, Furfaro F, Parisi F (2010) Querying and repairing inconsistent numerical databases. *ACM Trans Database Syst* 35(2):14:1–14:50
51. Franconi E, Palma AL, Leone N, Perri S, Scarcello F (2001) Census data repair: a challenging application of disjunctive logic programming. In: Nieuwenhuis R, Voronkov A (eds) Proceedings of the 8th international conference logic for programming, artificial intelligence and reasoning (LPAR 2001), LNCS, vol. 2250. Springer, Berlin, pp 561–578
52. Galhardas H, Florescu D, Shasha D, Simon E, Saita CA (2001) Declarative data cleaning: language, model, and algorithms. In: Proceedings of the 27th international conference on very large data bases (VLDB 2001). Morgan Kaufmann, Orlando, pp 371–380
53. Gardezi J, Bertossi L (2012) Query rewriting using datalog for duplicate resolution. In: Barceló P, Pichler R (eds) Proceedings of the second international workshop on datalog in academia and industry, Datalog 2.0, LNCS, vol. 7494. Springer, Berlin, pp 86–98
54. Gardezi J, Bertossi L (2012) Tractable cases of clean query answering under entity resolution via matching dependencies. In: Hüllermeier E, Link S, Foerster T, Seeger B (eds) Proceedings of the 6th international conference scalable uncertainty management (SUM 2012), LNCS, vol. 7520. Springer, Berlin, pp 180–193

55. Gardezi J, Bertossi L, Kiringa I (2012) Matching dependencies: semantics and query answering. *Front Comput Sci* 6(3):278–292
56. Giannotti F, Pedreschi D, Saccà D, Zaniolo C (1991) Non-determinism in deductive databases. In: Delobel C, Kifer M, Masunaga Y (eds) *Proceedings of the second international conference on deductive and object-oriented databases (DOOD 1991)*, LNCS, vol. 566. Springer, Berlin, pp 129–146
57. Golab L, Karloff HJ, Korn F, Srivastava D, Yu B (2008) On generating near-optimal tableaux for conditional functional dependencies. *VLDB* 1(1):376–390
58. Golab L, Korn F, Srivastava D (2011) Efficient and effective analysis of data quality using pattern tableaux. *IEEE Data Eng Bull* 34(3):26–33
59. Greco G, Greco S, Zumpano E (2003) A logical framework for querying and repairing inconsistent databases. *IEEE Trans Knowl Data Eng* 15(6):1389–1408
60. Gupta A, Mumick IS (1995) Maintenance of materialized views: problems, techniques and applications. *IEEE Q Bull Data Eng* 18(2):3–18
61. Kolahi S, Lakshmanan LVS (2009) On approximating optimum repairs for functional dependency violations. In: *Proceedings of the 12th international conference on database theory (ICDT'09)*. ACM, New York, pp 53–62
62. Lenzerini M (2002) Data integration: a theoretical perspective. In: *Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems (PODS 2002)*. ACM, New York, pp 233–246
63. Lenzerini M (2012) Ontology-based data management. In: Freire J, Suciu D (eds) *Proceedings of the 6th Alberto Mendelzon international workshop on foundations of data management (AMW 2012)*, CEUR workshop proceedings, CEUR-WS.org, vol. 866, pp 12–15
64. Malaki A, Bertossi L, Rizzolo F (2012) Multidimensional contexts for data quality assessment. In: Freire J, Suciu D (eds) *Proceedings of the 6th Alberto Mendelzon international workshop on foundations of data management (AMW 2012)*, CEUR workshop proceedings, CEUR-WS.org, vol. 866, pp 196–209
65. Motro A, Anokhin P (2006) Fusionplex: resolution of data inconsistencies in the integration of heterogeneous information sources. *Inform Fusion* 7(2):176–196
66. Naumann F, Herschel M (2010) An introduction to duplicate detection. *Synthesis lectures on data management*. Morgan & Claypool Publishers, San Rafael
67. Nicolas JM (1982) Logic for improving integrity checking in relational data bases. *Acta Inform* 18(3):227–253
68. Rahm E, Do HH (2000) Data cleaning: problems and current approaches. *IEEE Data Eng Bull* 23(4):3–13
69. Saïs F, Pernelle N, Rousset MC (2007) L2R: a logical method for reference reconciliation. In: *Proceedings of the 22nd AAAI conference on artificial intelligence (AAAI 2007)*. AAAI Press, pp 329–334
70. Talbut J (2013) A practical guide to entity resolution with OYSTER. In: Sadiq S (ed) *Handbook on data quality management*. Springer, Berlin, Handbook Series (this volume)
71. ten Cate B, Fontaine G, Kolaitis PG (2012) On the data complexity of consistent query answering. In: Deutsch A (ed) *Proceedings of the 15th international conference on database theory (ICDT 2012)*. ACM, New York, pp 22–33
72. Türker C, Gertz M (2001) Semantic integrity support in SQL:1999 and commercial (object-)relational database management systems. *VLDB J* 10(4):241–269
73. Whang SE, Benjelloun O, Garcia-Molina H (2009) Generic entity resolution with negative rules. *VLDB J* 18(6):1261–1277
74. Wijesn J (2005) Database repairing using updates. *ACM Trans Database Syst* 30(3):722–768
75. Yakout M, Elmagarmid AK, Neville J, Ouzzani M, Ilyas IF (2011) Guided data repair. *VLDB* 4(5):279–289

# Linking Records in Complex Context

Pei Li and Andrea Maurino

**Abstract** There are different kinds of information present in a data set that can be utilized for record linkage activities: attributes, context, relationships, etc. In this chapter, we focus on techniques that enable record linkage in so-called complex context, which includes data sets with hierarchical relations, data sets that contain temporal information, and data sets that are extracted from the Web. For each method, we describe the problem to be solved and use a motivating example to demonstrate the challenges and intuitions of the work. We then present an overview of the approaches, followed by more detailed explanation of some key ideas, together with examples.

## 1 Introduction

*Record linkage* takes a set of records as input and discovers which records refer to the same real-world entity. It plays an important role in data integration, data aggregation, and personal information management and has been extensively studied in recent years (see [1, 2] for recent surveys). Existing techniques typically proceed in two steps: the first step compares similarity between each pair of records, deciding if they match or do not match; the second step clusters the records accordingly, with the goal that records in the same cluster refer to the same real-world entity and records in different clusters refer to different ones.

---

P. Li (✉)

Department of Informatics, University of Zurich, Binzmuehlestrasse 14, CH-8050 Zurich, Switzerland

e-mail: [peili@ifi.uzh.ch](mailto:peili@ifi.uzh.ch)

A. Maurino

Department of Informatics, Systems and Communication, University of Milano, Bicocca, Viale Sarca 336/14, 20126 Milano, Italy

e-mail: [maurino@disco.unimib.it](mailto:maurino@disco.unimib.it)

For record-similarity computation, existing works can be divided into three categories: *classification-based* approaches [3] classify a pair of records as *match*, *unmatch*, and *maybe*; *distance-based* approaches [4] apply distance metrics to compute similarity of each attribute and take the weighted sum as the record similarity; *rule-based* approaches [5] apply domain knowledge to match records.

For record clustering, there exists a wealth of literature on clustering algorithms for record linkage [6]. These algorithms may apply the transitive rule [6] and efficiently perform clustering by a single scan of record pairs (e.g., *partition algorithm* [5]), may iteratively specify seeds of clusters and assign vertexes to the seeds (e.g., *ricochet algorithm* [7]), and may perform clustering by solving an optimization problem (e.g., *correlation clustering* [8], *Markov clustering* [9], and *cut clustering* [10]).

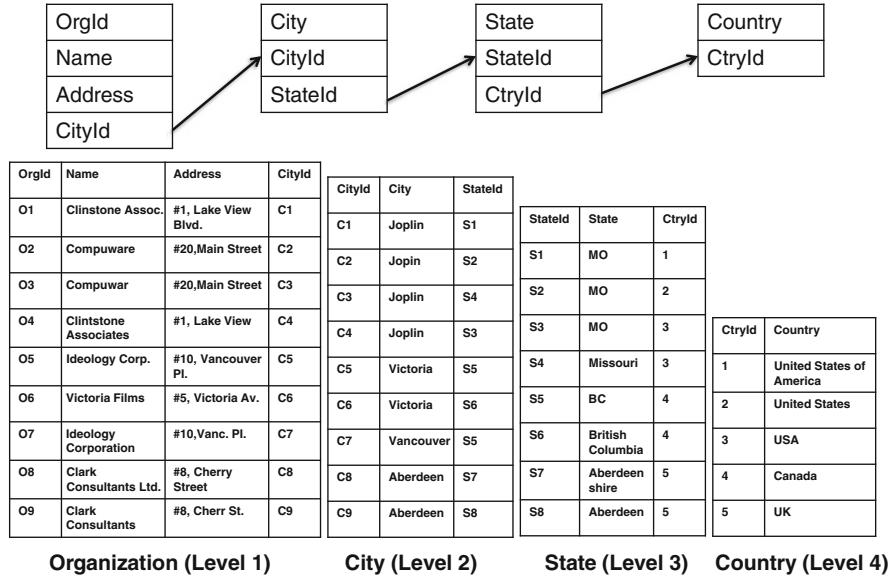
The above approaches for matching and clustering utilize different kinds of information to enable record linkage. The traditional approach is to compare the values in attributes of two entity descriptions (references) to decide whether they co-refer (i.e., refer to the same real-world entity), for all pairs of entity references. These are called feature-based similarity (FBS) methods [3, 5]. In addition, there has been a number of interesting techniques developed that enhance the traditional techniques by being able to utilize certain types of context to improve the quality. These techniques can be categorized as follows:

- Techniques [11–13] that utilize directly linked entities as additional information when computing record similarity (Sects. 2–4)
- Techniques [14, 15] that can analyze inter-entity relationships of references (Sect. 5)
- Techniques that take into account other information, such as information from the Web [16, 17] (Sect. 6), temporal information [18] (Sect. 7), and spatial information [19]
- Techniques [20] that analyze meta-information about the data set, specified by analysts
- Techniques [21–23] that take into account the dependence among the co-reference decisions and no longer make pair-wise matching decisions independently

In the following sections, we present several techniques that take into account complex context information for record linkage. For each method, we first describe the problem to be solved and use a motivating example to demonstrate the challenges and intuitions of the work; we then present an overview of the approaches, followed by more detailed explanation of some key ideas with examples.

## 2 Hierarchical-Structure-Based Approaches

The problem of record linkage is one of the major problems in the broad area of data cleaning and data quality. Record linkage is hard because it is caused by several types of errors like typographical errors, and *equivalence errors*—



**Fig. 1** A customer database in Example 1

different (nonunique and nonstandard) representations of the same logical value. For instance, a user may enter “WA, United States” or “Wash., USA” for “WA, United States of America.” Equivalence errors in product tables (“winxp pro” for “Windows XP Professional”) are different from those encountered in bibliographic tables (“VLDB” for “very large databases”), etc.

Previous approaches for record linkage either are addressed by building sets of rules, which are domain-dependent, or rely on textual similarity functions (e.g., edit distance or cosine metric), predicting that two tuples whose textual similarity is greater than a prespecified similarity threshold are duplicates. However, using these functions to detect duplicates due to equivalence errors requires that the threshold be dropped low enough, resulting in a large number of *false positives*. For instance, tuple pairs with values “USSR” and “United States” in the country attribute are also likely to be declared duplicates if we were to detect “US” and “United States” as duplicates using textual similarity. To solve the problem, Ananthkrishna et al. proposed a domain-independent, hierarchical-structure-based approach [11], called Delphi, which relies on hierarchies to detect equivalence errors in each relation and to reduce the number of false positives. We use the following example to demonstrate the motivations.

*Example 1.* Consider a customer information database that consists of four relations: organization, city, state, and country in Fig. 1. Suppose we want to disambiguate “United States” and “USA” in country relation. We observe that USA and United States *co-occur* through state MO in state relation. In general, country

tuples are associated with sets of state values. The degree of overlap between sets associated with two countries is a measure of co-occurrence between them and can be used to detect duplicates (e.g., USA and United States).

The notion of co-occurrence can also reduce the number of false positives. Consider “USA” and “UK” in Fig. 1. We might incorrectly deduce that they are duplicates because of their similarity. However, we observe that the state sets of USA and UK in state relation, {MO, Missouri} and {Aberdeen, Aberdeenshire}, respectively, are disjoint. This indicates that the USA and UK are unlikely to be duplicates.

Before describing Delphi algorithm in detail, we first give basic definition of the problem. Given a set of relations in a database, a *dimensional hierarchy* consists of a chain of relations linked by key-foreign key dependencies. An *entity* described by the hierarchy consists of a chain of tuples (one from each relation), each of which joins with the tuple from its parent relation. Two entities in a dimensional hierarchy are duplicates if corresponding pairs of tuples in each relation of the hierarchy either match exactly or are duplicates (according to duplicate-detection functions at each level). For example, two entities in Fig. 1 are duplicates if the respective pairs of country, state, city, and organization tuples of the two entities are duplicates.

The main idea of Delphi is to exploit knowledge from already processed relations. This usage requires us to process a parent relation in the hierarchy before processing its child. As we move down the hierarchy, the reduction in the number of comparisons is significant. To this end, we adopt a *top-down traversal* of the hierarchy. Specifically, the algorithm proceeds as follows:

1. Starting from the top level of the hierarchy and for each relation that has not been processed:
  - (a) Group relations below current relation into clusters of tuples.
  - (b) Prune each cluster according to properties of similarity functions eliminating tuples that cannot be duplicates.
  - (c) Compare pairs of tuples within each group by (1) textual similarity between two tuples and (2) co-occurrence similarity between the children sets of the tuples.
  - (d) Combine the results of the above two measures, and compare against a given threshold or a set of thresholds to decide if two tuples are duplicates.
  - (e) Dynamically update the thresholds to structural characteristics of different clusters.
2. The algorithm terminates when all relations in the hierarchy are processed.

Experimental results show that Delphi works well in real-world data sets that observe dimensional hierarchies. However, when relations in the databases are not restricted to *one-to-many* (or *many-to-one*) relationships, assumptions of duplicates in this work do not hold.

### 3 Iterative Record Linkage

As stated in the previous section, traditional approaches for record linkage often use a similarity measure that compares tuples' attribute values; tuples with similarity scores above a certain threshold are declared to be matches. While this method can perform quite well in many domains, particularly domains where there is not a large amount of noise in the data, in some domains looking only at tuple values is not enough. By also examining the context of tuple, i.e., the other tuples to which it is linked, we can come up with a more accurate linkage decision. But this additional accuracy comes at a price. In order to correctly find all duplicates, we may need to make multiple passes over the data; as linkages are discovered, they may in turn allow us to discover additional linkages.

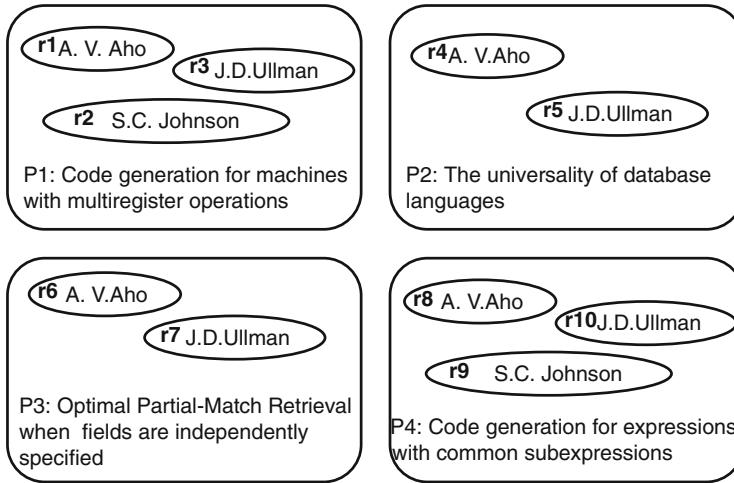
To illustrate the power and feasibility of making use of join information when comparing records, Bhattacharya and Getoor [12] propose an iterative record linkage approach that exploits both attribute similarity and the similarity of linked objects. Different from other methods that take advantage of context information, the authors do not assume that the linked objects have already been deduplicated. They consider cases where the links are among entities of the same type so that when two records are determined to refer to the same real-world entity, this may in turn allow additional inferences and make the deduplication process iterative.

We next take an example of author resolution problem to demonstrate the motivations and key ideas of the proposed approach.

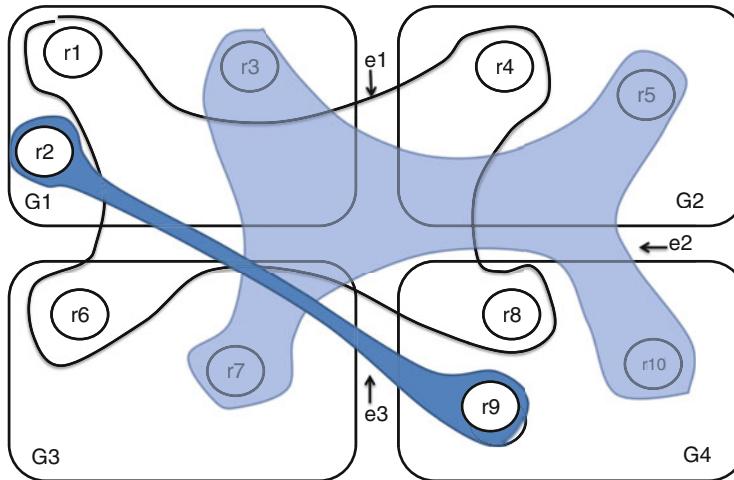
*Example 2.* Consider the author reference example in Fig. 2, where there are four paper references, each with a title and author references. We need to find out which author references refer to the same real-world individuals. Figure 3 shows the correct results.

We may first decide that all of the Aho references refer to the same individual, because Aho is an unusual last name. This corresponds to identifying  $r_1, r_4, r_6$ , and  $r_8$  as duplicates. Given such information, we may consolidate the  $r_5$  and  $r_7$  Ullman references, since Aho has collaborated in two papers ( $P_2, P_3$ ) with authors named Ullman, and it is highly likely that the two Ullman references ( $r_5$  and  $r_7$ ) refer to the same individual. We may also consolidate the other Ullman references ( $r_5$  and  $r_{10}$ ) with lower confidence. Once we have resolved the references of Aho and Ullman, for similar reason we may conclude that the two Johnson references ( $r_2$  and  $r_9$  in paper  $P_1$  and  $P_4$ , respectively) refer to the same person.

We next present formal statement of the record linkage problem before describing the proposed iterative algorithm. Given a collection of references  $R = \{r_1, r_2, \dots, r_n\}, n > 0$ , each reference  $r$  corresponds to a unique entity,  $E(r) \in \{e_1, e_2, \dots, e_k\}, k > 0$ , and conversely  $R(e) = \{r_i | E(r_i) = e\}$ . The references may be partitioned into groups  $G = \{g_1, g_2, \dots, g_m\}, m > 0$ , a group. The task is, given  $R$  and  $G$ , to correctly determine both the entities and the mapping from references to entities.



**Fig. 2** Author/paper resolution problem in Example 2



**Fig. 3** Ground truth of author references in Example 2

*Example 3.* Consider the author resolution problem in Example 2. Given author references  $r_1 - r_{10}$  and papers  $P_1 - P_4$ , the set of author references for each paper is considered as a group. For example,  $P_1$  defines the group  $g_1 = \{r_1 - r_3\}$ , where  $r_1 = \text{A.V. Aho}$ ,  $r_2 = \text{S.C. Johnson}$ , and  $r_3 = \text{J.D. Ullman}$ . The correct resolution results lead to three author entities, where  $e_1 = \{r_1, r_4, r_6, r_8\}$ ,  $e_2 = \{r_3, r_5, r_7, r_{10}\}$ , and  $e_3 = \{r_2, r_9\}$ .

The key idea of the proposed approach is to take into account both attribute similarity and the links between entities. The attribute similarity of two references

measures the similarity of the attributes of two references. In addition, the approach compares two groups of references by looking at which references are currently known/believed to be duplicates. This notion of similarity is bound to the current set of duplicates and will change as new duplicates are found. Next we describe the details of the approach.

Duplication is defined by a weighted combination of the attribute distance of the references and distance between their groups. The definition of duplicates is recursive in that the distance between references is tied to the current set of duplicates. We focus on the group similarity of references. For a reference  $r$ ,  $G(r)$  is all the groups that  $r$  or its duplicates occur in. The similarity of two groups is defined as the ratio of the number of duplicates they share and the length of the longer group. Specifically,

$$\text{sim}(g_1, g_2) = \frac{|\text{common}(g_1, g_2)|}{\max\{|g_1|, |g_2|\}}, \quad (1)$$

$$\text{common}(g_1, g_2) = \{(r_1, r_2) \mid \text{dup}(r_1, r_2), r_1 \in g_1, r_2 \in g_2\}. \quad (2)$$

where  $g_1$  and  $g_2$  are two groups and  $\text{dup}(r_1, r_2)$  is considered as duplicates. The distance of two groups  $g_1$  and  $g_2$  is defined as  $1 - \text{sim}(g_1, g_2)$ . Consider the similarity of two groups  $g_2$  and  $g_3$  in Example 2, where  $g_2 = \{r_4, r_5\}$  and  $g_3 = \{r_6, r_7\}$ . Assume that  $r_4$  and  $r_6$  are detected as duplicates; thus,  $\text{sim}(g_2, g_3) = \frac{1}{2} = 0.5$ . The similarity of a group  $g$  and a group set  $G$  is computed as  $\max_{g' \in G} \text{sim}(g, g')$ . Consider  $g_1 = \{r_1 - r_3\}$  and the group set  $G$  of reference  $r_5$  in Example 2. Assume we have detected that  $r_5, r_7$  are duplicates, and thus,  $G = \{g_2, g_3\}$ ; then  $\text{sim}(g_1, G) = \max\{\text{sim}(g_1, g_2), \text{sim}(g_1, g_3)\} = \frac{1}{3}$ . Finally, the similarity of two group sets  $G$  and  $G'$  is the average similarity between groups in  $G$  and groups in  $G'$ .

As new duplicates are discovered, the similarity of group sets of references is changing, potentially leading to the discovery of more duplicates. Current set of duplicates are represented as clusters, each of which is associated by the set of groups that its references occur in. Specifically the algorithm proceeds as follows:

1. Initialize a clustering by merging references that are highly similar.
2. For each iteration, choose the candidate cluster pairs that are likely to be the same entity. Evaluate similarity of the candidate pairs, merge the clusters that are most similar, and update their group sets.
3. The algorithm terminates when the candidate set is exhausted.

## 4 Record Linkage in Complex Information Spaces

The above-presented approaches focus on resolving references of a *single* class that has a fair number of attributes (e.g., research publications). In this section, we consider complex information spaces: the references belong to *multiple* related classes, and each reference may have very few attribute values. A prime example

**Table 1** Article references in Example 4

ID	Title	Pages	Authored by	Published in
$a_1$	Distributed query processing in a relational database system	169–180	$p_1 - p_3$	$c_1$
$a_2$	Distributed query processing in a relational database system	169–180	$p_4 - p_6$	$c_2$

**Table 2** Person references in Example 4

ID	Title	Pages	Coauthor	Email contact
$p_1$	Robert S. Epstein	Null	$p_2, p_3$	Null
$p_2$	Michael Stonebraker	Null	$p_1, p_3$	Null
$p_3$	Eugene Wong	Null	$p_1, p_2$	Null
$p_4$	Epstein, R.S.	Null	$p_5, p_6$	Null
$p_5$	Stonebraker, M.	Null	$p_4, p_6$	Null
$p_6$	Wong, E.	Null	$p_4, p_5$	Null
$p_7$	Epstein Wong	eugene@berkeley.edu	Null	$p_8$
$p_8$	Null	stonebraker@csail.mit.edu	Null	$p_7$
$p_9$	Mike	stonebraker@csail.mit.edu	Null	Null

**Table 3** Conference references in Example 4

ID	Name	Year	Location
$c_1$	ACM Conference on Management of Data	1978	Austin, Texas
$c_2$	ACM SIGMOD	1978	Null

of such a space is personal information management (PIM), where the goal is to provide a coherent view of all the information on one's desktop. The work of Dong et al. [13] is the first to resolve different references in complex information spaces, where references belong to *multiple* related classes. The proposed algorithm has three principal features: (1) it exploits the associations between references for similarity comparison; (2) it propagates information between linkage decisions to accumulate positive and negative evidences; (3) it gradually enriches references by merging attribute values. We next demonstrate the motivation and intuitions of the work before describing the details of the proposed algorithm.

*Example 4.* Consider resolving a set of references extracted from a personal data set in a personal information management, which contains **person**, **article**, and **conference** in Tables 1–3. The **article** references  $a_1$  and  $a_2$ , **person** references  $p_1$  to  $p_6$ , and **conference** references  $c_1$  and  $c_2$  are extracted from two BibTex items. The other two **person** references,  $p_7$  to  $p_9$ , are extracted from emails. The ideal resolving result for the above references are as follows: **Article** references  $a_1, a_2$  refer to the same article; **Person** references  $p_2, p_5, p_8, p_9$  refer to the same person;  $p_3, p_6, p_7$  refer to the same person; and **Conference** references  $c_1, c_2$  refer to the same conference.

Existing reference resolution algorithms may fall short for applications such as PIM for the following reasons:

- It is often the case that each reference includes very limited information, i.e., contains values for only a few attributes. For example, a `person` reference often has values for only one or two attributes. In Example 4, references  $p_5$  and  $p_8$  even do not have any attributes in common.
- Some attribute values are multivalued, so the fact that two attribute values are different does not imply that the two references refer to different real-world entities. For example, two `person` references with completely different email addresses may refer to the same person.

The key idea of the proposed approach is to exploit the richness of the information space at hand. We next illustrate the main features with the above motivating example and then describe the details of the algorithm.

**Exploiting Context Information:** The algorithm makes extensive use of *context information* (the associations between references) to provide evidence for linkage decisions. For example, given two references of persons, the algorithm considers their coauthors and email contacts, to help in deciding whether to merge them. In Example 4,  $p_5$  has coauthored articles with  $p_6$ , and  $p_6$  has email correspondences with  $p_7$ . If we decide that  $p_6$  and  $p_7$  are the same person, we obtain additional evidence that may lead us to merge  $p_5$  and  $p_8$ . In addition, the algorithm compares values of different attributes. For example, the name “Stonebraker, M.” and the email address “stonebraker@scail.mit.edu” are closely related. This observation provides positive evidence for merging references  $p_5$  and  $p_8$ .

**Reconciliation Propagation:** The algorithm *propagates* information between linkage decision for different pairs of references. For example, when the algorithm decides to merge two papers, it obtains additional evidence for merging the person references to their authors. This in turn can further increase the confidence in reconciling other papers authored by the resolved persons. In Example 4, article references  $a_1$  and  $a_2$  share the same title and similar authors and that they appeared in similar conferences and pages in the proceedings; we decide to merge them. Presumably an article has a unique set of authors, so the reconciliation of  $a_1$  and  $a_2$  implies that the authors  $p_1$  and  $p_4$ ,  $p_2$  and  $p_5$ , and  $p_3$  and  $p_6$  should be reconciled respectively. Similarly, conference references  $c_1$  and  $c_2$  are reconciled.

**Reference Enrichment:** The algorithm addresses the lack of information in each reference by *reference enrichment*. For example, when the algorithm merges two person references, it gathers the different representations of the person’s name, collects her different email addresses, and enlarges her list of coauthors and email contacts. This enriched reference can later be merged with other references. Consider the person references  $p_5$ ,  $p_8$ , and  $p_9$  in Example 4. After  $p_8$  and  $p_9$  are merged, we can aggregate their information and know that “mike” and “Stonebraker, M.” share the same first name initial and contact the same person by email correspondence or coauthoring. This additional information will enable us to merge  $p_5$ ,  $p_8$ , and  $p_9$ .

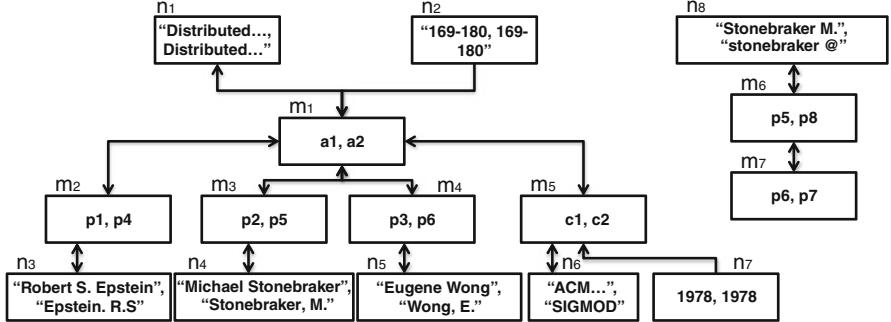


Fig. 4 The dependency graph for references in Example 4

The proposed approach is facilitated by an *independency graph*, where a node in the graph represents the similarity between a pair of references and an edge presents the dependency between a pair of similarities, e.g., the similarity of a pair of references depends on the similarity of their respective attributes and vice versa.

Figure 4 shows the dependency graph  $G$  for references in Example 4. For each pair of references of the same class, there is a node  $m$  in  $G$ ; for each pair of attribute values of two references, there is a node  $n$  in  $G$  and an edge between  $m$  and  $n$ . Node  $m$  is a *real-valued neighbor* of  $n$ , if the similarity of  $n$  depends on the actual similarity value of node  $m$ ;  $m$  is called  $n$ 's *weak-boolean-valued neighbor*, if the reconciliation of  $m$ 's references only increases the similarity score of  $n$  but does not directly imply reconciliation; and  $m$  is  $n$ 's *strong-boolean-valued neighbor*, if the reconciliation of  $m$ 's two references implies that the two references in  $n$  should also be reconciled. For example, the similarity of papers  $a_1$  and  $a_2$  is represented by the node  $m_1$ ; it is dependent on the similarity of the titles (represented by  $n_1$ ), the pages ( $n_2$ ), the authors ( $m_2, m_3$ , and  $m_4$ ), and the conferences ( $m_5$ ). Note that there does not exist an edge from  $m_5$  to  $n_7$ , because the similarity of years is predetermined and is independent of the similarity of any conferences. Also note that there is no node  $(p_1, p_5)$ , because their name attributes have very different values. Consider references  $r_5$  and  $r_8$ . Given that  $p_5$  has coauthored with  $p_6$  and  $p_8$  has email correspondence with  $p_7$ , there is a node  $m_7 = (p_6, p_7)$  and an edge between  $m_6$  and  $m_7$ . Note that  $m_6$  does not have a neighbor  $(p_4, p_7)$ , because  $p_4$  and  $p_7$  do not have any similar attributes and so the node  $(p_4, p_7)$  does not exist.

The proposed algorithm is based on propagating similarity decisions from node to node in the graph. For example, after we decide to reconcile articles  $a_1$  with  $a_2$ , we should reconcile their associated conferences  $c_1$  and  $c_2$  and further trigger recomputation of the similarities of other papers that mention the conferences  $c_1, c_2$ , etc. Given that the dependency graph has captured all the dependencies between similarities, it guides the recomputation process. The algorithm selects the recomputation order to improve efficiency based on the following heuristics:

- The similarity for a node is computed only if the scores of its incoming neighbors have all been computed, unless there exist mutual dependencies. For example,

we compare two articles only after comparing their associated authors and conferences (or journals).

- When a node is merged, its outgoing neighbors is first considered for recomputation.

The algorithm proceeds by maintaining a queue of active nodes. Initially, the queue contains all reference-similarity nodes, and it satisfies the property that a node always precedes its outgoing real-valued neighbors if there does not exist mutual dependencies. At every iteration, we compute the similarity score for the top node in the queue. If we activate its outgoing *real-valued* or *weak-boolean-valued* neighbors, then we insert them at the end of the queue. If we activate its *strong-boolean-valued* neighbors, we insert them in front of the queue.

*Example 5.* Consider the dependency graph in Fig. 4. Initially, the queue contains nodes  $\{m_5, m_4, m_3, m_2, m_1\}$ , and nodes  $n_1, n_2$ , and  $n_7$  are merged. We then compute the similarity of  $m_5, m_4, m_3, m_2$ , and  $m_1$  in succession. When papers  $a_1$  and  $a_2$  are merged, we insert  $m_2, m_3, m_4$ , and  $m_5$  back to the front of the queue. Note that  $n_2$  is not added back both because it is not an outgoing neighbor of  $m_1$  and because it already has similarity score 1. Next, we consider  $m_5$  and decide to merge  $c_1$  and  $c_2$ , so we insert its strong-boolean-valued neighbor,  $n_6$ , in the front of the queue. This process continues until the queue is empty.

Another important aspect of the algorithm is to enrich the references in the process of reconciliation. For example, if  $r_1$  has email address `stonebraker@csail.mit.edu` and  $r_2$  has email address `stonebraker@mit.edu`, the real-world person object actually has both email addresses. When we compute the similarity between  $r_1$  and another reference  $r_3$ , we compare both email addresses with the email of  $r_3$ , and choose the one with a higher similarity. Specifically, after we decide to merge references  $r_1$  and  $r_2$ , we search for all references  $r_3$  such that there exist nodes  $m = (r_1, r_3)$  and  $n = (r_2, r_3)$ . We proceed to remove  $n$  from the graph in the following steps:

1. Connect all neighbors (incoming and outgoing neighbors) of  $n$  with  $m$  while preserving the direction of the edges.
2. Remove node  $n$  and its associated edges from the dependency graph and from the queue.
3. If  $m$  gets new incoming neighbors and is not active in the queue, we insert  $m$  at the end of the queue; similarly,  $n$ 's neighbors that get new incoming neighbors and are not active are inserted at the end of the queue.

## 5 Relationship-Based Approaches

Contextual information can be grouped in two types: linked objects that are real-world entities directly related to the duplicate records and inter-object relationships, which include direct and indirect connections between entities. In this section, we

**Table 4** Records in Example 6: (a) author records; (b) paper records

ID	Name	Affiliation	ID	Title	Coauthors
$A_1$	Dave White	Intel	$P_1$	Databases...	John Black, Don White
$A_2$	Don Whilte	CMU	$P_2$	Multimedia...	Sue Grey, <b>D. White</b>
$A_3$	Susan Grey	MIT	$P_3$	Title 3	Dave White
$A_4$	John Black	MIT	$P_4$	Title 5	Don White, Joe Brown
$A_5$	Joe Brown	Unknown	$P_5$	Title 6	Joe Brown, Liz Pink
$A_6$	Liz Pink	Unknown	$P_6$	Title 7	Liz Pink, <b>D. White</b>

(a)

(b)

consider inter-object relationships. For instance, “D. White” might be used to refer to an author in the context of a particular publication. This publication might also refer to different authors, which can be linked to their affiliated organizations, etc., forming chains of relationships among entities. Such knowledge can be exploited alongside attribute-based similarity resulting in improved accuracy of disambiguation.

To this end, Kalashnikov et al. [24] propose a domain-independent entity resolution approach, referred to as relationship-based data cleaning (RelDC), which systematically exploits relationships among entities to improve disambiguation quality. We demonstrate the motivation of the approach by the following example.

*Example 6.* Consider a database about *authors* and *publications* in Table 4. The goal is to identify for each author reference in each paper the correct author it refers to.

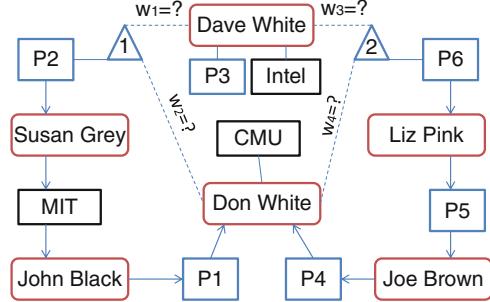
We first consider existing feature-based similarity (FBS) techniques to compare each author reference in papers with values in `name` attribute in authors. This would allow us to resolve almost every author reference in the above example. For instance, such methods would identify that “Sue Grey” reference in  $P_2$  refers to  $A_4$  (“Susan Grey”). The only exception will be “D. White” references in  $P_2$  and  $P_6$ : “D. White” could match either  $A_1$  (“Dave White”) or  $A_2$  (“Don White”).

We next consider additional attributes to disambiguate “D. White.” For instance, the titles of papers  $P_1$  and  $P_2$  might be similar, while titles of  $P_2$  and  $P_3$  might not, suggesting that “D. White” of  $P_2$  is indeed “Don White” of paper  $P_1$ . We next show that exploiting relationships among entities can further improve disambiguation when we are unable to disambiguate the references using title (or other attributes).

First, we observe that author “Don White” has coauthored a paper ( $P_1$ ) with “John Black” who is at MIT, while the author ‘Dave White’ does not have any coauthored papers with authors at MIT. We can use this observation to disambiguate between the two authors. In particular, since the coauthor of “D. White” in  $P_2$  is “Susan Grey” of MIT, there is a higher likelihood that the author “D. White” in  $P_2$  is “Don White.” The reason is that the data suggests a connection between author “Don White” with MIT and an absence of it between “Dave White” and MIT.

Second, we observe that author “Don White” has coauthored a paper ( $P_4$ ) with “Joe Brown” who in turn has coauthored a paper with “Liz Pink.” In contrast, author “Dave White” has not coauthored any papers with either “Liz Pink” or “Joe Brown.”

**Fig. 5** Entity-relationship graph for records in Example 6



Since “Liz Pink” is a coauthor of  $P_6$ , there is a higher likelihood that “D. White” in  $P_6$  refers to author ‘Don White’ compared to author “Dave White.” The reason is that often coauthor networks from groups/clusters of authors that do related research and may publish with each other. The data suggests that “Don White,” “Joe Brown,” and “Liz Pink” are part of the cluster, while “Dave White” is not.

To generalize the above analysis, we consider the database as a graph of interconnected entities (modeled as nodes in the graph) linked to each other via relationships (modeled as edges in the graph). Figure 5 illustrates the entity-relationship graph corresponding to Example 6.

We can disambiguate “D. White” in  $P_2$  according to the path between node “Don White” and  $P_2$  in Fig. 5, i.e.,  $P_2 \rightarrow$  “Susan Grey”  $\rightarrow$  “MIT”  $\rightarrow$  “John Black”  $\rightarrow P_1 \rightarrow$  “Don White.” Similarly, we can disambiguate “D. White” in  $P_6$  by the following path:  $P_6 \rightarrow$  “Liz Pink”  $\rightarrow P_5 \rightarrow$  “Joe Brown”  $\rightarrow P_4 \rightarrow$  “Don White.” Both paths suggest that the “D. White” references probably correspond to the author “Don White.” However, in principle “D. White” could also be “Dave White,” since there could also be paths between  $P_2(P_6)$  and “Dave White.” To determine if ‘D. White’ is “Don White” or “Dave White,” we need to measure whether “Don White” or “Dave White” is more strongly connected to  $P_2(P_6)$ .

Before explaining the algorithm, we first give several notations. In Fig. 5, we assign for each edge a weight, a real value in  $[0,1]$ , which reflects the degree of confidence the relationship exists. By default, all weights are equal to 1. In Fig. 5, there are two triangle nodes 1 and 2 that are *choice nodes*. For example,  $P_2$  is connected to ‘Dave White’ and “Don White” by choice node 1, meaning that ‘D. White’ in  $P_2$  can be either “Dave White” (with probability of  $w_1$ ) or “Don White” (with probability of  $w_2$ ), where  $w_1 + w_2 = 1$ . The key idea of RelDC is the notion of *connection strength* between two entities  $x$  and  $y$  (denoted by  $c(x, y)$ ) that captures how strongly  $x$  and  $y$  are connected to each other through relationships. Specifically, the algorithm proceeds as follows:

1. *Compute connection strengths:* For each node pair  $x$  and  $y$ , compute the connection strength  $c(x, y)$ , which is the probability to reach  $y$  from  $x$  by following only  $L$ -short simple paths such that the probability to follow an edge is proportional to the weight of the edge. A path is  $L$ -short if its length is no greater

than parameter  $L$ . The probability to follow path  $p$  is computed as the probability to follow each of its edges.

2. *Determine equations for edge weights:* Using the equation from step 1, determine a set of equations that relate edge weights to each other. In particular, we use the strategy where weights are proportional to the corresponding connection strengths.
3. *Compute weights:* Solve the set of equations from step 2.
4. *Resolve references:* Interpret the weights computed in step 3 as well as attribute-based similarity to resolve references.

Given Example 6, we first compute connection strengths as follows:

- $c_1 = c(P_2, \text{"DaveWhite"}) = c(P_2 \rightarrow \text{Susan} \rightarrow \text{MIT} \rightarrow \text{John} \rightarrow P_1 \rightarrow \text{Don} \rightarrow P_4 \rightarrow \text{Joe} \rightarrow P_5 \rightarrow \text{Liz} \rightarrow P_6 \rightarrow \text{"2"} \rightarrow \text{"Dave White"}) = \frac{w_3}{2}$ .
- $c_2 = c(P_2, \text{"DonWhite"}) = c(P_2 \rightarrow \text{Susan} \rightarrow \text{MIT} \rightarrow \text{John} \rightarrow P_1 \rightarrow \text{"DonWhite"}) = 1$ .
- $c_3 = c(P_6, \text{"DaveWhite"}) = \frac{w_1}{2}$ .
- $c_4 = c(P_6, \text{"DonWhite"}) = 1$ .

We next determine equations for the above weights:

- $w_1 = c_1 / (c_1 + c_2) = \frac{w_3}{2} / (1 + \frac{w_3}{2})$ .
- $w_2 = c_2 / (c_1 + c_2) = 1 / (1 + \frac{w_3}{2})$ .
- $w_1 = c_3 / (c_3 + c_4) = \frac{w_1}{2} / (1 + \frac{w_1}{2})$ .
- $w_1 = c_4 / (c_3 + c_4) = 1 / (1 + \frac{w_1}{2})$ .

Note that all weights should be in [0,1]; we then compute weights in the above equations as follows:  $w_1 = 0$ ,  $w_2 = 1$ ,  $w_3 = 0$ , and  $w_4 = 1$ . Thus, “D. White” in  $P_2$  and  $P_6$  refers to “Don White.”

## 6 Web-Based Approaches

World Wide Web (WWW) search engines are commonly used for learning about real-world entities, such as people. In such cases, users search the name of the target entities in search engines to obtain a set of Web pages that contain that name. However, ambiguity in names typically causes the search results to contain false positives. For example, if we want to know about a “George Bush” other than the former US president, many pages about the former president are returned in the search results, which may be problematic.

Typical approaches of disambiguating people’s names [25, 26] are to define similarities between documents based on features extracted from the documents and cluster Web pages returned by search engines using the similarity. Most effective features include *named entities (NEs)*, *compound keywords (CKWs)*, and *URLs*, which are considered as *strong features*. The problem with such features is that they show high precision but low recall. Solutions to improve recall can be either to

reduce the threshold for document similarities or to use *weak features* for document similarity calculation, such as single words. Both approaches improve recall with big sacrifice on precision.

To overcome the aforementioned problem, Yoshida et al. [17] propose a two-stage approach that disambiguates person names in Web search results. The system uses named entities, compound keywords, and URLs as features for document similarity calculation, which typically show high precision, and applies bootstrapping to improve recall. Specifically, the results of the first stage are used to extract features used in the second-stage clustering. In the following texts, we present formal statement of the problem and highlight the motivations and key techniques of the work, followed by a summary of its experimental results.

We denote a query (target person name) by  $q$  and the set of Web pages obtained by inputting query  $q$  to a search engine by  $\mathcal{P} = \{d_1, d_2, \dots, d_k\}, k > 0$ , where a Web page  $d$  has at least one string  $q$ . We assume that  $q$  on the same page refers to the same entity. Person-name disambiguation is to cluster documents in  $\mathcal{P}$  where each cluster refers to a single entity.

The main idea of the two-stage algorithm is to cluster documents only using *strong features* in the first stage and revises clustering results by using weak features in the second stage. For example, if a computer scientist and a baseball player share the same name, words like *memory* and *algorithm* are reliable weak features for the former, and words like *ball* and *batting* are reliable weak features for the latter. The algorithm proceeds as follows.

**First-stage clustering** Clusters retrieved documents on the basis of strong feature similarities, including named entities, compound keywords, and URLs. We next describe these features and then explain how to calculate their similarity, followed by the clustering method we use in the first stage.

- Named entity features include person names, organization names, and place names, which typically represent real-world entities related to the person.
- Compound keyword features are combinations of single nouns that are used as term units more frequently than others. They are extracted by considering frequencies of term units.
- URL features include URLs extracted from each document and the URL of the document itself.

We use overlap coefficient to calculate named entity and compound keyword features as follows:

$$\text{Overlap}(d_i, d_j) = \frac{|\bar{f}_i \cap \bar{f}_j|}{\max(\min(|\bar{f}_i|, |\bar{f}_j|), \theta_{overlap})} \quad (3)$$

where  $\bar{f}_i$  and  $\bar{f}_j$  are sets of features extracted from document  $d_i$  and  $d_j$ , respectively.  $\theta_{overlap}$  is a threshold determined by training data. URL similarity, denoted by  $\text{sim}_{URL}$ , equals to 1 if  $d_i$  links to  $d_j$  (vice versa) and is calculated by  $\text{overlap}(d_i, d_j)$

otherwise. The overall similarity of two documents is computed as the weighted sum to these feature similarities.

Once document similarity is calculated, we use standard hierarchical agglomerative clustering (HAC) algorithm for clustering. Specifically, the algorithm starts from one-by-one clustering and iteratively merges the most similar cluster pairs if their similarities are above a predefined threshold.

**Second-stage clustering** Exploits bootstrapping approach where the algorithm regards the first-stage clusters as seed instances, finds weak feature related to them, and finds new instances (new documents) by using the weak features (as extraction patterns). The algorithm proceeds as follows:

1. Generate a feature-document matrix  $P$  given the set  $D$  of documents.
2. For each iteration of bootstrapping:
  - (a) Given current clustering, calculate a feature-cluster matrix  $R_F$  from  $P$  and document-cluster matrix  $R_D$ , i.e.,  $R_F = \frac{1}{|D|} PR_D$ .
  - (b) Given  $P$  and  $R_F$ , calculate a document-cluster matrix  $R'_D$  for next iteration, where  $R'_D = \frac{1}{|F|} PR_F$ .
3. For each document, find the optimal cluster it belongs to.

The algorithm is tested on WePS benchmark data set that consists of 30 names, each of which has 150 pages. Comparison with several baseline approaches and WePS top systems shows that the algorithm outperforms all of them.

## 7 Temporal Record Linkage

Many data sets contain temporal records over a long period of time; each record is associated with a time stamp and describes some aspects of a real-world entity at that particular time (e.g., author information in DBLP). In such cases, we often wish to identify records that describe the same entity over time and so be able to enable interesting longitudinal data analysis. However, existing record linkage techniques ignore the temporal information and can fall short for temporal data. Li et al. [18] propose a formal model to link temporal records that refer to the same real-world entities. We first demonstrate the motivations and intuitions of the work by the following example.

*Example 7.* Consider records that describe paper authors in Table 5; each record is derived from a publication record at DBLP (we may skip some coauthors for space reason). These records describe three real-world persons:  $r_1$  describes  $E_1$ , *Xin Dong*, who was at *R. Polytechnic* in 1991;  $r_2-r_6$  describe  $E_2$ , *Xin Luna Dong*, who moved from *Univ of Washington* to *AT&T Labs*; and  $r_7-r_{12}$  describe  $E_3$ , *Dong Xin*, who moved from *Univ of Illinois* to *Microsoft Research*.

If we require high similarity on both **name** and **affiliation**, we may split entities  $E_2$  and  $E_3$ , as records for each of them can have different values for **affiliation**. If

**Table 5** Records from DBLP

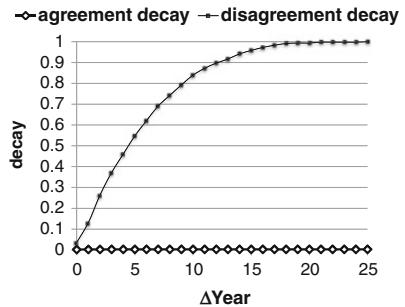
ID	Name	Affiliation	Coauthors	Year
$r_1$	Xin Dong	R. Polytechnic Institute	Wozny	1991
$r_2$	Xin Dong	Univ of Washington	Halevy, Tatarinov	2004
$r_3$	Xin Dong	Univ of Washington	Halevy	2005
$r_4$	Xin Luna Dong	Univ of Washington	Halevy, Yu	2007
$r_5$	Xin Luna Dong	AT&T Labs-Research	Das Sarma, Halevy	2009
$r_6$	Xin Luna Dong	AT&T Labs-Research	Naumann	2010
$r_7$	Dong Xin	Univ of Illinois	Han, Wah	2004
$r_8$	Dong Xin	Univ of Illinois	Wah	2007
$r_9$	Dong Xin	Microsoft Research	Wu, Han	2008
$r_{10}$	Dong Xin	Univ of Illinois	Ling, He	2009
$r_{11}$	Dong Xin	Microsoft Research	Chaudhuri, Ganti	2009
$r_{12}$	Dong Xin	Microsoft Research	Ganti	2010

we require only high similarity of `name`, we may merge  $E_1$  with  $E_2$ , as they share the same name, and may even merge all of the three entities.

Despite the challenges, temporal information does present additional evidence for linkage.

- Record values typically transition *smoothly*. In the motivating example, person  $E_3$  moved to a new affiliation in 2008 but still had similar coauthors from previous years.
- Record values seldom change *erratically*. In our example,  $r_2, r_3, r_7, r_8$ , and  $r_{10}$  are very unlikely to refer to the same person, as a person rarely moves between two affiliations back and forth over many years. (However, this can happen around transition time; for example, entity  $E_3$  has a paper with the old affiliation information after he moved to a new affiliation, as shown by record  $r_{10}$ .)
- In case we have a fairly complete data set such as DBLP, records that refer to the same real-world entity often (but not necessarily) observe *continuity*; for example, one is less confident that  $r_1$  and  $r_2-r_6$  refer to the same person given the big time gap between them. Exploring such evidence would require a global view of the records with the time factor in mind.

We next present the overview of proposed approach that leverages temporal information with linkage. First, when computing record similarity, traditional linkage techniques reward high-value similarity and penalize low-value similarity. However, as time elapses, values of a particular entity may evolve; for example, a researcher may change affiliation, email, and even name over time (see entities  $E_2$  and  $E_3$  in Example 7). Meanwhile, different entities are more likely to share the same value(s) with a long time gap; for example, it is more likely that we observe two persons with the same name within 30 years than at the same time. Thus, the concept of *decay* is defined, with which we can reduce penalty for value disagreement and reduce reward for value agreement over a long period.

**Fig. 6** Address decay curves

Second, when clustering records according to record similarity, traditional techniques do not consider time order of the records. However, time order can often provide important clues. In Example 7, records  $r_2-r_4$  and  $r_5-r_6$  may refer to the same person even though the decayed similarity between  $r_4$  and  $r_6$  is low, because the time period of  $r_2-r_4$  (year 2004–2007) and that of  $r_5-r_6$  (year 2009–2010) does not overlap; on the other hand, records  $r_2-r_4$  and  $r_7, r_8$ , and  $r_{10}$  are very likely to refer to different persons even though the decayed similarity between  $r_2$  and  $r_{10}$  is high, because the records interleave and their occurrence periods highly overlap. Therefore, temporal clustering algorithms are proposed that consider time order of records and can further improve linkage results. In the following texts, we describe the two main components of the proposed approach.

As time goes by, the value of an entity may evolve; for example, entity  $E_2$  in Example 7 was at *UW* from 2004 to 2007 and moved to *AT&T Labs* afterwards. Thus, different values for a single-valued attribute over a long period of time should not be considered as strong indicator of referring to different entities. The notion of *disagreement decay* is defined to capture this intuition. On the other hand, as time goes by, we are more likely to observe two entities with the same attribute value; for example, in Example 7, entity  $E_1$  occurred in 1991 and  $E_2$  occurred in 2004–2010, and they share the same name. Thus, the same value over a long period of time should not be considered as strong indicator of referring to the same entity. Accordingly, the notion of *agreement decay* is defined.

Figure 6 shows the curves of disagreement decay and agreement decay on attribute address learned from a European patent data set.

We observe that (1) the disagreement decay increases from 0 to 1 when time elapses, showing that two records differing in affiliation over a long time is not a strong indicator of referring to different entities; (2) the agreement decay is close to 0 everywhere, showing that in this data set, sharing the same address is a strong indicator of referring to the same entity even over a long time; and (3) even when  $\Delta t = 0$ , neither the disagreement nor the agreement decay is exactly 0 meaning that even at the same time, address match does not correspond to record match and vice versa.

We next describe how we apply decay in record-similarity computation. When computing similarity between two records with a big time gap, we often wish to

reduce the penalty if they have different values and reduce the reward if they share the same value. Thus, we assign weights to the attributes according to the decay; the lower the weight, the less important is an attribute in record-similarity computation, so the less penalty for value disagreement or the less reward for value agreement. This weight is decided both by the time gap and by the similarity between the values (to decide whether to apply agreement or disagreement decay). Once attribute weights are determined, the record similarity is computed as weighted sum of attribute similarity.

We next present three proposed clustering methods, all processing the records in increasing time order. *Early binding* makes eager decisions and merges a record with an already created cluster once it computes a high similarity between them. *Late binding* compares a record with each already created cluster and keeps the probability and makes clustering decision at the end. *Adjusted binding* is applied after early binding or late binding and improves over them by comparing a record also with clusters created later and adjusting the clustering results.

**Early binding** Considers the records in time order; for each record it eagerly creates its own cluster or merges it with an already created cluster. In particular, consider record  $r$  and already created clusters  $C_1, \dots, C_n$ . The algorithm proceeds in three steps:

1. Compute the similarity between  $r$  and each  $C_i, i \in [1, n]$ .
2. Choose the cluster  $C$  with the highest similarity. Merge  $r$  with  $C$  if  $\text{sim}(r, C) > \theta$ , where  $\theta$  is a threshold indicating high similarity; create a new cluster  $C_{n+1}$  for  $r$  otherwise.
3. Update signature for the cluster with  $r$  accordingly.

**Late Binding:** Instead of making eager decisions and comparing a record with a cluster based on such eager decisions, late binding keeps all evidence, considers them in record-cluster comparison, and makes a global decision at the end. Late binding is facilitated by a bipartite graph  $(N_R, N_C, E)$ , where each node in  $N_R$  represents a record, each node in  $N_C$  represents a cluster, and each edge  $(n_r, n_C) \in E$  is marked with the probability that record  $r$  belongs to cluster  $C$ . Late binding clusters the records in two stages: first, *evidence collection* creates the bipartite graph and computes the weight for each edge; then, *decision making* removes edges such that each record belongs to a single cluster.

**Adjusted Binding:** Neither early binding nor late binding compares a record with a cluster created later. However, evidence from later records may fix early errors. Adjusted binding allows comparison between a record and clusters that are created later. It can start with the results from either early or late binding and iteratively adjust the clustering as follows:

1. *Initialization:* Set the initial assignment as the result of early or late binding.
2. *Estimation (E-step):* Compute the similarity of each record-cluster pair and normalize the similarities as in late binding.

3. *Maximization* (M-step): Choose the clustering with the maximum probability, as in late binding.
4. *Termination*: Repeat E-step and M-step until the results converge or oscillate.

## 8 Summary

Due to heterogeneous schemas, possible errors in data sets, and faulty update processes, traditional record linkage techniques may fall short for many cases. Additional contextual information in the data thus needs to be considered to improve the results for linkage. In this chapter we have described several techniques proposed for solving record linkage problem in complex context. Specifically, we consider entities connected to the duplicate tuples in hierarchical structures, directed linked entities that allow us to iteratively discover additional linkages as initial linkages are discovered, complex information spaces where references to the same real-world entity belong to multiple related classes, and inter-object relationships, which include direct and indirect connections between entities. In addition, we also consider Web information and temporal information for linkage. There are still continuing challenges in exploiting complex contextual information for record linkage, including combining multiple contextual information for linkage, such as combining temporal information with spatial information to achieve better results and to enhance record linkage with information in more complicated frameworks, such as information from crowdsourcing.

## References

1. Elmagarmid AK, Ipeirotis PG, Verykios VS (2007) Duplicate record detection: a survey. *IEEE Trans Knowl Data Eng* 19(1):1–16
2. Koudas N, Sarawagi S, Srivastava D (2006) Record linkage: similarity measures and algorithms. In: Proceedings of the ACM SIGMOD international conference on management of data, Chicago, 27–29 June 2006
3. Fellegi IP, Sunter AB (1969) A theory for record linkage. *J Am Stat Assoc* 64(328):1183–1210
4. Dey D (2008) Entity matching in heterogeneous databases: a logistic regression approach. *Decis Support Syst* 44:740–747
5. Hernandez MA, Stolfo SJ (1995) The merge/purge problem for large databases. In: Proceedings of the ACM SIGMOD international conference on management of data, San Jose, 22–25 May 1995
6. Hassanzadeh O, Chiang F, Lee HC, Miller RJ (2009) Framework for evaluating clustering algorithms in duplicate detection. In: Proceedings of 35th international conference on very large data bases (VLDB 2009), Lyon, 24–28 August 2009
7. Wijaya DT, Bressan S (2009) Ricochet: a family of unconstrained algorithms for graph clustering. In: Proceedings of the 14th international conference on database systems for advanced applications (DASFAA 2009). Lecture Notes in Computer Science, vol. 5463, Brisbane, 21–23 April 2009. Springer, Berlin

8. Bansal N, Blum A, Chawla S (2002) Correlation clustering. In: Proceedings of the 43rd annual IEEE symposium on foundations of computer science (FOCS'02), Vancouver, 16–19 November 2002
9. van Dongen S (2000) Graph clustering by flow simulation. PhD thesis, University of Utrecht
10. Flake G, Tarjan R, Tsioutsiouliklis K (2004) Graph clustering and minimum cut trees. Internet Math 1:385–408
11. Ananthakrishna R, Chaudhuri S, Ganti V (2002) Eliminating fuzzy duplicates in data warehouses. In: Proceedings of 28th international conference on very large data bases (VLDB 2002), Hong Kong, 20–23 August 2002
12. Bhattacharya I, Getoor L (2004) Iterative record linkage for cleaning and integration. In: Proceedings 9th ACM SIGMOD workshop on research issues in data mining and knowledge discovery, Maison de la Chimie, Paris, 13 June 2004
13. Dong X, Halevy A, Madhavan J (2005) Reference reconciliation in complex information spaces. In: Proceedings of the ACM SIGMOD international conference on management of data (SIGMOD'05), Baltimore, 13–16 June 2005
14. Chen Z, Kalashnikov DV, Mehrotra S (2005) Exploiting relationships for object consolidation. In: Proceedings of the 2nd international ACM SIGMOD workshop on Information quality in information systems (IQIS'05), Baltimore, 17 June 2005
15. Malin B (2005) Unsupervised name disambiguation via social network similarity. In: Proceedings of workshop on link analysis, counterterrorism, and security, Newport Beach, 23 April 2005
16. Lee T, Wang Z, Wang H, Hwang S-W (2011) Web scale taxonomy cleansing. In: Proceedings of 37th international conference on very large data bases (VLDB 2011), vol. 4, no. 12, Seattle, 29 August–3 September 2011, pp 1295–1306
17. Yoshida M, Ikeda M, Ono S, Sato I, Nakagawa H (2010) Person name disambiguation by bootstrapping. In Proceedings of the 33rd international ACM SIGIR conference on research and development in information retrieval (SIGIR'10), Geneva, 19–23 July 2010
18. Li P, Dong XL, Maurino A, Srivastava D (2011) Linking temporal records. In: Proceedings of 37th international conference on very large data bases (VLDB 2011), vol. 4, no. 11, Seattle, 29 August–3 September 2011, pp 956–967
19. Sehgal V, Getoor L, Viechnicki PD (2006) Entity resolution in geospatial data integration. In: Proceedings of the 14th ACM international symposium on advances in geographic information systems (GIS'06), Arlington, 10–11 November 2006
20. Councill IG, Li H, Zhuang Z, Debnath S, Bolelli L, Lee WC, Sivasubramaniam A, Giles CL (2006) Learning metadata from the evidence in an on-line citation matching scheme. In: Proceedings of the 6th ACM/IEEE-CS joint conference on digital libraries (JCDL'06), Chapel Hill, 11–15 June 2006
21. Bhattacharya I, Getoor L (2005) Relational clustering for multi-type entity resolution. In: Proceedings of the 4th international workshop on multi-relational mining (MRDM'05), Chicago, 21 August 2005
22. McCallum AK, Wellner B (2003) Toward conditional models of identity uncertainty with application to proper noun coreference. In: Proceedings of IJCAI-03 workshop on information integration on the web (IIWeb-03), Acapulco, 9–10 August 2003
23. Domingos P (2004) Multi-relational record linkage. In: Proceedings of the KDD-2004 workshop on multi-relational data mining, Seattle, 22 August 2004
24. Kalashnikov DV, Mehrotra S, Chen Z (2005) Exploiting relationships for domain-independent data cleaning. In: Proceedings of 2005 SIAM international conference on data mining (SIAM SDM'05), Newport Beach, 21–23 April 2005
25. Artiles J, Gonzalo J, Sekine S (2007) The semeval-2007 weps evaluation: establishing a benchmark for the web people search task. In: Proceedings of SemEval-2007: 4th international workshop on semantic evaluations, Prague, 23–30 June 2007
26. Artiles J, Gonzalo J, Sekine S (2009) Weps 2 evaluation campaign: overview of the web people search clustering task. In: Proceedings of WePS-2 second web people search evaluation workshop, Madrid, 21 April 2009

# A Practical Guide to Entity Resolution with OYSTER

John R. Talburt and Yinle Zhou

**Abstract** This chapter discusses the concepts and methods of entity resolution (ER) and how they can be applied in practice to eliminate redundant data records and support master data management programs. The chapter is organized into two main parts. The first part discusses the components of ER with particular emphasis approximate matching algorithms and the activities that comprise identity information management. The second part provides a step-by-step guide to build an ER process including data profiling, data preparation, identity attribute selection, rule development, ER algorithm considerations, deciding on an identity management strategy, results analysis, and rule refinement. Each step in the process is illustrated with an actual example using the OYSTER open-source, entity resolution system.

## 1 Introduction

In the list of the ten root conditions of data quality problems given by Lee, Pipino, Funk, and Wang [1], the first condition listed is “multiple sources of the same information.” Similarly, [2] emphasizes the need to control data redundancy. As with most things, data redundancy has both positive and negative aspects. The positive aspect is that while one source may be missing an important item of information about an entity, another one of the sources does have that item of information. Therefore, by combining the information from all of the sources, a more complete picture of the entity emerges. For example, if the entity in question is a customer of a business, the multiple sources could be his or her purchases across multiple sales channels over an extended period of time. An analysis of this information could help the business more effectively focus its marketing to each customer on an individual

---

J.R. Talburt (✉) · Y. Zhou  
University of Arkansas at Little Rock, Little Rock, AR, USA  
e-mail: [jrtalburt@ualr.edu](mailto:jrtalburt@ualr.edu); [yxzhou@ualr.edu](mailto:yxzhou@ualr.edu)

basis. The goal is to have the so-called 360-degree view of each customer, a state that is considered the foundation of customer relationship management (CRM) [3].

The negative aspects of using multiple sources of the same information are all too familiar to anyone experienced in data management. Instead of being complementary, multiple sources that are not properly integrated leave information about the same entity disconnected and isolated. When this happens, it can give the impression that there are more entities that really exist in the real world. Even when records about the same entity from different sources are brought together properly, they sometimes provide inconsistent and conflicting information about the entity. These problems can be especially damaging when the entities represent the master data of the organization. *Master data* are those records that represent the most critical entities in an organization, each of which has its own separate identity, such as employees, customers, sales, vendors, and product lines.

From a data quality standpoint, multiple sources of information about the same entities represent the problem of maintaining *entity identity integrity*. Entity identity integrity is one of the basic tenets of data quality that applies to the representation of a given domain of real-world entities in an information system [4]. Entity identity integrity requires that:

- Each real-world entity in the domain has one and only one representation in the information system.
- Distinct real-world entities have distinct representations in the information system.

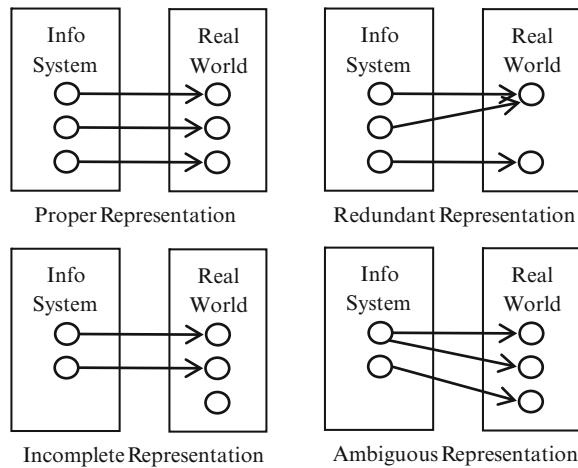
Entity identity integrity has also been described as *proper representation* [5]. Figure 1 illustrates the concept of proper representation and the typical ways in which it fails to hold.

One of the most common problems caused by multiple sources of information is the relationship shown in Fig. 1 and described as redundant representation. This is the condition that two or more information system records are referring to the same real-world entity. *Entity resolution* (ER) is the process of determining whether two records in an information system that reference real-world objects are referring to the same object or to different objects [6]. Entity resolution is a key process in establishing entity identity integrity and a precursor to entity-based data integration and master data management (MDM). Records that specifically reference real-world objects are often called *references*.

The identity of an entity is a set of attribute values for that entity along with a set of distinct rules that allow that entity to be distinguished from all other entities of the same class in a given context [7]. In the research literature, other terms are used to describe the ER process, such as “record linkage” [8, 9], “de-duplication” [10], “object identification” [11], and “disambiguation” [12].

The terms “record linkage” and the “record matching problem” are most commonly used in the statistics community to describe ER [8, 13–18]. In the database community, the problem is referred to most commonly as “merge-purge” [19], data “de-duplication” [10, 20], and “instance identification” [21].

**Fig. 1** Errors in proper representation



In the artificial intelligence (AI) community, this problem is described as “database hardening” [22] and “name matching” [23]. The terms “co-reference resolution,” “identity uncertainty,” and “duplicate detection” are also commonly used to refer to the same task [24]. The term entity resolution (ER) first appeared in publications by researchers at the Stanford InfoLab led by Hector Garcia-Molina and is defined as the process of identifying and merging records judged to represent the same real-world entity [25].

ER is a key task in data integration where different systems or data sources provide information for a common set of entities. The commercial application of ER has its roots in customer relationship management (CRM) where it is often referred to as customer data integration (CDI) [3, 26]. ER also plays an important role in price comparison search for online shopping [27], epidemiological statistics for biomedical research [28], and data mining for counter terrorism [29].

The need for accurate and efficient ER is now being driven by the need to share entity-based information across independently governed organizations in areas such as education, health care, and law enforcement. For example, ER systems are at the heart of the growing number of health care information exchanges (HIEs) that attempt to improve patient care, further clinical research, and contain rising costs by sharing information across patients, providers, and payers [30].

## 2 Components of Entity Resolution

From a high-level perspective, ER comprises five major components or subprocesses [6]. These five major ER components are:

1. Entity reference extraction: The process of locating and collecting entity references from unstructured information such as free-form text or images and converting them to structured entity references.

2. Entity reference preparation: The process of assessing and improving the structured entity references prior to making ER decisions, a process that may involve data profiling, standardization, data cleansing, data enhancement, and other data quality techniques.
3. Entity reference resolution: The process of systematically comparing entity references and deciding whether they refer to the same, or to different, entities. The decisions are based on *identity rules* (*matching rules*) that measure the degree of similarity between the attribute values of two references or, in some cases, a pattern of relationships among several references.
4. Entity identity information management: The process of creating and maintaining *entity identity structures* (EIS) that store and accumulate identity information over time, a process that allows references to entities to be labeled with the same identifier over time.
5. Entity analytics: The process of investigating associations among distinct but often related entities. One of the most common is called *householding*. *Customer householding* is where a business attempts to understand which of their customer share a common address or family relationship. *Corporate householding* tries to identify and connect separate business entities that have legal or contractual relationships such as subsidiaries of a larger corporation or franchisees of a national or international brand. Entity analytics is also an important tool in fraud detection and criminal investigations as a way to expose connections among different individuals acting in concert or the same individual using different aliases and personas.

## 2.1 Entity Reference Extraction

Entity references for ER processes usually need be collected and extracted from unstructured textual data (UTD) [31], such as email, spreadsheets, medical records, documents, and reports, and placed in a structured format like a database or XML (Extensible Markup Language). The process of filling out predefined structured summaries from text documents is called information extraction (IE) [32]. Entity reference extraction or IE has been researched extensively. Different natural language processing (NLP) technologies are used in IE, and the research on NLP supports and affects the directions in IE [33]. Many IE algorithms and systems are developed to extract information from traditional UTD, such as newspaper and journal articles, and medical records. Data that was originally stored in paper format could be scanned and translated to electronic format by optical character recognition (OCR) techniques. In recent years IE from web pages written in HTML has become a popular topic. HTML is considered as semi-structured document that can be more easily parsed than free-format text documents [34]. Named entity recognition (NER) is a special type of IE and involves identifying references to particular kinds of objects such as names of people, companies, and locations [35].

The two most commonly used types of IE systems are rule-based systems and machine learning-based systems [36]. Rule-based systems operate with manually

developed information extraction rules that detect certain patterns, for example, Suiseki system [37]. Machine learning-based systems automatically learn pattern-based extraction rules using supervised machine learning methods [38]. For example, in PAPIER [39] patterns are expressed in an enhanced regular expression language, and a bottom-up relational rule learner is used to induce rules from a corpus of labeled training examples; Webfoot and CRYSTAL [40] use rules in the first stage of an extraction flow to identify basic features such as capitalized words or phone numbers. Then they use these basic features as input to various machine learning algorithms that implement the rest of the extraction process. There are many other IE systems being developed and used both academically and practically. Entity reference extraction, or the broader concept of IE, is still a vibrant research area and plays important role in ER processes.

## 2.2 *Entity Reference Preparation*

As has been commonly acknowledged in the discipline of information quality, the second ER activity of reference preparation has historically been neglected or simply taken for granted. Some of the most recent research has been on the impact of poor data quality on ER and data mining outcomes, as well as information system processes in general.

Data, after being extracted and stored in a structured format, usually reflects various data quality problems. The application of profiling, standardization, data cleansing, and other data quality techniques to structured entity references prior to the start of the resolution process is necessary. The entity reference extraction and preparation discussed is also sometimes described as ETL (extraction, transformation, loading) [41].

As the discipline of information and data quality becomes more mature, data preparation is receiving more attention in both research and commercial practice literature. Several operations commonly performed in a data preparation process are [42]:

- Data profiling: A process whereby one examines the data available in an existing database or flat file and collects statistics and information about that data [43].
- Parsing: Locates, identifies, and isolates data elements embedded in a character string. For example, parsing the full name “John A. Doe” to a first name “John,” middle name “A,” and a last name “Doe.” Another term to describe a similar operation is segmentation [44].
- Standardization: A process of transforming the information represented in certain fields into a common and agreed upon set of formats or values. For example, the same address can be represented in many different ways, like “123 Oak St.” and “123 Oak Street,” but under standardization rules, only one would be acceptable (i.e., in standard form).
- Encoding: Translating incoming data from one character encoding to another, e.g., ASCII to EBCDIC character encoding.

- Conversion: Transforming one data type to another data type, e.g., binary integer into a numeric character string.
- Enhancement: Adding information not in the original reference based on information in the reference, e.g., adding longitude and latitude coordinates based on street address.

There are many other operations that can be performed during the data preparation phase, like correction, bucketing of continuous values, and bursting (splitting) multi-entity records. Different data quality software vendors offer systems that incorporate some or all of these capabilities. Some of the leading commercial data quality software vendors are SAS DataFlux [45], Informatica [46], and IBM [47]. There are also many open-source data quality software products as well [48]. Some of these are Talend Open Studio and Open Profiler [49], Ataccama DQ Analyzer [50], Pentaho Kettle [51], and SQL Power Architect [52].

### ***2.3 Entity Reference Resolution***

Entity reference resolution is the process of resolving (deciding) whether two references to real-world entities are to the same entity or to different entities. To make these decisions, ER systems commonly employ two techniques that work in concert: direct matching and transitive equivalence [6].

Direct matching is the determination of equivalence between two references based on the degree of similarity between the values of corresponding identity attributes. There are two general types: exact (deterministic) and probabilistic (fuzzy) matching.

Exact match simply means that the attributes' values must be identical. The requirement that all identity attributes must be an exact match is called deterministic matching. When attribute values are represented as character strings, exact matching is problematic because even the slightest variance prevents the values from matching. For example, the name “John Doe” is not an exact match for “JOHN DOE” or even “John Doe” where there are two spaces between the names instead of one space. The success of deterministic matching schemes relies on extensive preparation activities, particularly standardization and cleaning.

Most large-scale ER applications employ some level of approximate match between attributes. Matching that allows for some attributes to only be similar rather than exact is called probabilistic matching or “fuzzy” matching. The algorithms that measure the degree of similarity between attribute values are called similarity functions [53]. Similarity functions that make these determinations have many different forms. The four basic types are:

- Numerical similarity functions
- Syntactic similarity functions
- Semantic similarity functions
- Phonetic similarity functions

### 2.3.1 Numerical Similarity Functions

In the case of numeric attributes, such as age, the similarity function is a matter of establishing the total amount of allowable difference between the two values. Approximate matching between numeric values is not always simple subtraction. For example, with date and time values, the difference requires more elaborate calculations that compute the difference as the number days, hours, or minutes between the values. This feature is built into many tools such as automated spreadsheets and databases, so that subtracting the YYYYMMDD-format date 20091231 from 20100101 yields the value 1 (day) rather than the mathematic difference of 8,870.

### 2.3.2 Syntactic Similarity Functions

Syntactic similarity functions are also called approximate string matching (ASM) [54] or string comparator metrics [18]. These functions perform string matching between patterns in a text where one or both of them have suffered some kind of (undesirable) corruption. Because there is no right or wrong definition of string similarity, many different ASM algorithms have been developed, each focusing on a particular aspect of string similarity. The choice of ASM algorithm depends heavily on the type and condition of the data. For example, there are many ASM algorithms for comparing English names, but these may not work well for names in other languages or other types of data such as product names or street names. Following is a discussion of some of the more commonly used ASM algorithms.

## Jaccard Coefficient

The Jaccard coefficient is a similarity measure that, in its most general form, compares two sets P and Q with the following formula:

$$\text{Jaccard}(P, Q) = \frac{|P \cap Q|}{|P \cup Q|}$$

Essentially, the Jaccard coefficient measures the fraction of the data that is shared between P and Q compared to all data available in the union of these two sets. For example, consider the case where P = “SEAN DOE” and Q = “JOHN DOE.” If the data are considered at the token (word) level, then their intersection is the word “DOE” and their union are the words “SEAN,” “JOHN,” and “DOE.” Therefore, the coefficient value would be 1/3 or 0.333. On the other hand if the data are considered at the non-blank character level, then their intersection would be the four letters “E,” “N,” “D,” and “O” and their union the eight letters “S,” “E,” “A,” “N,” “J,” “O,” “H,” and “D.” In this case the Jaccard coefficient would be 4/8 or 0.500.

The advantage of the Jaccard coefficient is that it is not sensitive to word order because it considers only whether a token or character exists in a string, not at which position [53].

## Levenshtein Distance

Levenshtein distance [55] measures similarity between two character strings as the minimum number of basic character operations required to transform one string into the other. Typically the allowable operations are inserting one character, deleting one character, and replacing (substituting) one character. Some versions of the algorithm also allow transposing two adjacent characters. Using these manipulations, the string “JIM” can be transformed into the string “JAMES” in 3 operations by starting with “JIM,” substituting “A” for “I” to give “JAM,” and then inserting “E” and “S” to give “JAMES.”

The Levenshtein distance can also be normalized so it returns a rating value between zero and one, where a rating value of one means that the strings are identical. Rating values less than one represent proportionally lesser degrees of similarities between the strings. If A and B represent two characters strings and L(A, B) represents their Levenshtein distance, then the normalized Levenshtein is calculated as

$$\text{Normalized Levenshtein} = 1 - \frac{L(A, B)}{\text{Max}\{\text{Length}(A), \text{Length}(B)\}}$$

Using this formula the normalized Levenshtein distance between the strings “JIM” and “JAMES” in the previous example is 1–3/5 or 0.4. Although widely used in practice for duplicate detection, the Levenshtein distance is not a suitable similarity measure when whole segments of a string differ, e.g., when one string is a prefix of the second string (“Professor John Doe” vs. “John Doe”) or when strings use abbreviations (“Peter J Miller” vs. “Peter John Miller”). These problems are primarily due to the fact that all edit operations have equal weight and that each character is considered individually [53].

## Smith-Waterman Distance

Smith-Waterman distance is a function that identifies the longest common sub-expression between two strings [56]. Although a mainstay of bioinformatics used for measuring the similarity of base pairs in DNA sequences, it has also been adapted to other domains including name matching. For instance, it determines that the longest common sub-expression between “Professor John Doe” and “John R Doe” is the string “John Doe.”

The Smith-Waterman distance divides the original strings into a prefix, common sub-expressions, and a suffix. Most implementations assign lower weights to the

insertion or deletion of prefix or suffix blocks than to the insertion or deletions of individual characters mid-sequence. Hence, in the Smith-Waterman distance, the existence of a prefix or a suffix is penalized less than in the Levenshtein distance [53].

However, the Smith-Waterman distance is not sufficient to lower the penalty of gaps within the string, e.g., due to abbreviations and missing words. Essentially, it does not reflect the case where several common sub-expressions, divided by nonmatching character sequences, exist. This problem is addressed by the affine gap distance [57] that allows edit operations, notably insertion and deletion of complete blocks within a string. Moreover, it assigns these block insertions and deletions less weight than to the insertion or deletion of the individual characters in a block. Consequently, the affine gap distance gives the possibility to penalize less the nonmatching blocks of characters within a string, which is beneficial when comparing strings that contain abbreviations or that are missing some tokens. For example, the affine gap distance between the pair of strings “JOHN DOE” and “JOHN M DOE” is essentially the same as the affine gap distance between the pair “JOHN DOE” and “JOHN MORRIS DOE,” whereas the Smith-Waterman distance for the second pair would be much larger than for the first pair.

## q-Gram Distance

The term *q-gram* (also called *n-gram*) refers to a subsequence of q items from a given sequence. There are several q-gram algorithms, but all of them focus on the ordering of the characters as a measure of similarity [58]. The most basic is the maximum q-gram similarity measure. Given two character strings, the maximum q-gram similarity measure is the ratio of the length of the longest common substring to the length of the longest one of the two strings being compared.

An extension to the maximum q-gram algorithm is the q-Gram Tetrahedral Ratio [59]. It takes into consideration all q-grams shared between two strings. In particular it is the ratio of the total number of characters in shared q-grams to the total number of characters in all possible q-grams.

## Jaro and Jaro-Winkler Distance

Another q-gram variant is the Jaro distance [60]. It considers the number of characters in common between two strings and also the number of adjacent character transpositions. If A and B represent two strings with at least one character in common, the Jaro similarity is given by the following formula:

$$J(A, B) = W_1 \cdot \frac{C}{L_A} + W_2 \cdot \frac{C}{L_B} + W_3 \cdot \frac{(C - T)}{C}$$

where:

- $W_1$ ,  $W_2$ , and  $W_3$  are the weights assigned to the string A, string B, and transpositions, respectively.
- $W_1 + W_2 + W_3 = 1$ .
- C is the number of common characters.
- T is the number of transpositions.
- $L_A$  and  $L_B$  are the lengths of the two strings.

For example, the strings “SHAKLER” and “SHAKEL” have 6 characters in common and one transposition of “LE” to “EL.” Hence:

$$J(\text{SHAKLER}, \text{SHAKEL}) = \frac{1}{3} \cdot \frac{6}{7} + \frac{1}{3} \cdot \frac{6}{6} + \frac{1}{3} \cdot \frac{(6-1)}{6} = 0.897$$

The Jaro-Winkler comparator [61, 62] is a modification of the Jaro comparator that gives additional weight to the agreements on the first four characters of the two strings. If N represents the number of the first four characters that agree position by position, then the Jaro-Winkler similarity is calculated as

$$W(A, B) = J(A, B) + 0.1 \cdot N \cdot (1.0 - J(A, B))$$

In the example of “SHAKLER” and “SHAKEL” given above, the value of N is 4; thus, the Jaro-Winkler distance of these two strings is calculated as

$$W(\text{"SHAKLER"}, \text{"SHAKEL"}) = 0.897 + 0.1 \cdot 4 \cdot (1.0 - 0.897) = 0.938$$

### 2.3.3 Semantic Similarity Functions

Semantic similarity functions are based on the meaning of the attribute value rather than syntactic similarity [63]. Perhaps the most common are based on name variants (so-called nicknames) used in a particular language or country. As discussed earlier “JIM” and “JAMES” are considered semantically equivalent English names even though their Levenshtein Normalized Rating for similarity is only 0.40 or 40 %. Semantic similarity functions are usually implemented as a lookup algorithm using predefined tables of equivalent names.

### 2.3.4 Phonetic Similarity Functions

Phonetic similarity functions take into the pronunciation of words in a particular language or country, and, like semantic similarity, it is primarily used for name matching. For example, in English, the names “KAREN” and “CARYN” are pronounced the same even though they are syntactically different with a Levenshtein similarity of only 0.60. Phonetic similarity functions often take the form of “hash”

functions that operate on the characters of the input string to create a new output string in which similar sounding characters or groups of characters are assigned the same symbol. The objective is to have two phonetically similar strings transform into the same output string called a “hash token.”

Perhaps the most commonly used phonetic similarity function for English is the Soundex algorithm [64, 65]. In the Soundex algorithm, the first letter is retained, but vowels are removed and groups of similar sounding consonants are replaced with the same numeric digit. For example, the names “PHILIP” and “PHILLIP” both create the hash token “P410.” The algorithm for generating the hash token is [18]:

1. Capitalize all letters and drop punctuation.
2. Remove the letters A, E, I, O, U, H, W, and Y after the first step.
3. Keep first letter but replace the other letters by digits according to the coding {B, F, P, V} replace with 1, {C, G, J, K, Q, S, X, Z} replace with 2, {D, T} replace with 3, {L} replace with 4, {M, N} replace with 5, and {R} replace with 6.
4. Replace consecutive sequences of the same digit with a single digit if the letters they represent were originally next together in the name or separated by H or W.
5. If the result is longer than 4 characters total, drop digits at the end to make it 4 characters long. If the result is fewer than 4 characters, add zeros at the end to make it 4 characters long.

The New York State Identification and Intelligence System Phonetic Code, commonly known as NYSIIS, is a phonetic algorithm devised in 1970 as part of the New York State Identification and Intelligence System. It features an accuracy increase of 2.7% over the traditional Soundex algorithm [66].

### 2.3.5 Efficiency Considerations in Similarity Analysis

In theory, for any given data, if every reference is compared to every other reference and pairs of references that match by one of the identity rules are assigned the same link, then the entire set of references will be resolved. However, for large datasets, this simplistic approach to entity reference resolution process can be computationally intensive and can result in unacceptably long processing times.

There are generally two ways to reduce the amount of time necessary to resolve a set of references. One is to use techniques that reduce the number of comparisons, and the other is to use multiple processors.

Perhaps the most common technique for reducing the number of comparisons is to partition the records into blocks, and only pairs of records in each block are compared. Obviously the strategy for blocking is to create the blocks of records in such a way that matching pairs of records are most likely to occur within the same block. For example, if the primary identity attribute is name, then the records might be divided into blocks where all of the records in each block have names starting with the same first letter. There is extensive research on how to optimize the performance of ER by using blocking [67, 68].

Even though blocking can help with performance, the biggest disadvantage of blocking as with any technique for reducing the number of comparisons is that they cannot guarantee that all valid matches will be found. There is always the possibility that some pairs of matching records will be partitioned into different blocks and never compared. Poorly designed blocking may result in low accuracy of the ER result.

Another technique for reducing the number of comparisons is indexing. In particular this technique is to build an inverted index so that all records that share a common attribute value can be quickly located. For example, if a record has a last name value of “DOE,” then the index could be used to look up all other records that also have the last name value of “DOE” as potential candidates for matching.

One problem with indexing on the exact value is that many matching rules may use approximate match instead of exact match. Thus, even though “PHILLIP” and “PHILIP” might be considered a match, a lookup in the index using the key “PHILLIP” would not return records with the name “PHILIP.” For this reason, some index methods will index the hash value of the attribute instead of the actual attribute value. Consider an index based on the Soundex hash of the name. In this case, both “PHILLIP” and “PHILIP” produce the same Soundex hash value of “P410.” Using this technique will help find variants that might have been missed by an exact value index, but at the same time, it will likely increase the number of records retrieved for matching, many of which may not meet the matching criteria.

One of the most commonly used techniques used in commercial practice is the sorted neighborhood method (SNM) [69]. The SNM, also called the sliding window technique, is a hybrid of the blocking and indexing techniques. SNM first sorts all the entities using a preselected key built from all or part of certain attribute values. It then slides a fix-sized window from the beginning to the end of the list. In each sliding window, the first reference is compared to the rest of the references within the same window to identify pairs with small distances. After that the window moves (slides) down by one reference and the process repeats until it reaches the end of the list. The sliding window acts as a blocking scheme with the premise that matching references are lexicographically similar so that they tend to be located near each other in sorted order, i.e., within the same window [70].

## 2.4 Entity Identity Information Management

Entity identity information management (EIIM) is the collection and management of identity information with the goal of sustaining entity identity integrity [71]. Entity identity integrity is one of the basic tenets of data quality that applies to the representation of a given domain of real-world entities in an information system. Entity identity integrity has also been described as proper representation [5]. Entity identity integrity requires that [4]:

- Each real-world entity in the domain has one and only one representation in the information system.
- Distinct real-world entities have distinct representations in the information system.

An ER process will consistently identify references to the same entity for a given dataset, but EIIM allows an ER system to consistently label references to the same entity with the same identifier across datasets processed at different times. It does this by creating an entity identity structure (EIS) that carries identity information forward from process to process. This gives an EIIM system the capability to create and assign persistent entity identifiers, i.e., entity identifiers that do not change from process to process. The consistent labeling of master data is also one of the goals of master data management (MDM). EIIM provides a technological underpinning for MDM, which also encompasses many nontechnical issues related to master data policy and governance.

There are several EIS models, but three of the most common are the:

- Survivor record EIS
- Attribute-based EIS
- Record-based EIS

Each model represents a design decision about how much identity information to preserve at the end of the record-linking process. The survivor model is simply a decision to keep one record from each cluster to be a representative of the entire cluster. Typically the record with the most complete information is selected. In this model, the EIS is the same as the record structure.

A variation of the survivor model is the exemplar model. Rather than choosing an actual record from the cluster, a surrogate record is created by selecting the most representative value for each attribute from the values provided by the records in the cluster. Because the values in the exemplar record can come from different records, it may not correspond to an actual record in the cluster.

The attribute-based EIS model creates a list-of-lists structure. For each identity attribute represented in a cluster, the EIS saves a list of all the distinct values of that attribute. For example, if one of the attributes is First Name and the records in the cluster only have the values “JAMES,” “JIM,” and “JIMMY” for this attribute, then the list of distinct values for this attribute would only have these three values even if some of these values occurred in more than one record in the cluster. The advantage of the attribute-based EIS is that it saves all of the distinct values for each of the identity attributes in the cluster. However, it does not preserve the record integrity of the cluster. For example, the First Name attribute list might have the value “JIM” and the Last Name attribute list might have the value “DOE,” but it does not provide any information as to whether these two values actually occurred together in any particular record of the cluster.

The record-based EIS model saves all of the values by simply saving all of the records, at least the identity attributes portion of the records, which formed the

cluster. It has the advantage of maintaining data provenance for the attribute values but clearly requires more storage than either the survivor or attribute-based models.

EIIM designers must be careful about how EIS models and ER algorithms are combined. This is further complicated by the way in which the attribute values in the EIS are mapped to the matching rules. A more complete discussion of these issues can be found in [71].

## 2.5 Entity-Relationship Analysis

Entity-relationship analysis is exploring the network of associations among different but related entities. Researchers often use graphical methodologies for entity-relationship analysis. The underlying dataset is viewed as an entity-relationship graph, wherein the nodes represent the entities in the dataset and the edges represent the relationships among the entities, e.g., [72, 73]. For any two entity representations, the co-reference decision is made not only based on the entity features or information derived from the context but also based on the inter-entity relationships, including indirect ones, that exist among the two representations [74].

For example, SCAN [75] is a structural clustering algorithm for networks, which is used for graph partitioning or network clustering. Similarly, [53] describe algorithms for data with complex relationships. The first type is hierarchical relationships, including those found in XML data. These algorithms not only use relationship descriptions to potentially improve the effectiveness of duplicate detection, but they also prune pair-wise comparisons based on relationships.

## 3 A Demonstration of the Entity Resolution Process with OYSTER

Entity resolution has its theoretical foundations in the *Fellegi-Sunter Theory of Record Linking* [8]. A formal definition and model of entity resolution was developed in the form of the *Stanford Entity Resolution Framework (SERF)* [76]. The SERF model has been further generalized to include identity information management from the EIIM model [71] and also viewed in an information quality context in the Algebraic Model of Entity Resolution [6].

Even though the theoretical models of ER are quite interesting and offer important guidance on the ER system design and operating efficiency, in this section we present the application of the theory to practical situations. The aim is to create a better understanding of the important concepts and principles of ER by following a series of examples within an actual entity resolution system acting on synthetic, but realistic, customer data.

The ER system used for the examples is an open-source system named OYSTER ([sourceforge.net/projects/oysterer/](http://sourceforge.net/projects/oysterer/)) written in Java. The reference sources and identity rules used to resolve the references are specified by the user in the form of XML scripts. One advantage of OYSTER over other open-source ER systems is its support for identity information management. The OYSTER EIS are also in the form of XML documents that define and retain the identity information captured during the ER process.

To set the context of the following examples, assume there is a company named ABC, Inc. trying to integrate and manage their customer data. These examples will refer to three (synthetic) customer data files named List\_A, List\_B, and List\_C. For the interested reader, these files can be downloaded online from the website [ualr.edu/eriq/downloads](http://ualr.edu/eriq/downloads).

### ***3.1 Description of the Files***

The three customer data files have different subsets of the overall set of identity attributes, and the records themselves are in different formats. List\_A contains 94,306 references and is in a comma-delimited file format with quotation marks used as text qualifiers. There are eight attributes in List\_A and they are:

- Unique record identifier (*RecID*)
- Customer name (*Name*)
- Customer street address (*Address*)
- City, state, and zip code of the street address (*CityStateZip*)
- Customer post office box address (*POBox*)
- City, state, and zip code of the post office box address (*POCityStateZip*)
- Customer Social Security Number (*SSN*)
- Customer data of birth (*DOB*)

A segment of List\_A is shown in Fig. 2.

The List\_B file contains 100,777 references and the records are in a pipe-delimited format without a text qualifier character. List\_B has ten attributes:

- Unique record identifier (*RecID*)
- Customer first name (*FirstName*)
- Customer last name (*LastName*)
- Street number of the customer's address (*StrNbr*)
- Street name of the customer's address (*Address1*)
- Second line of customer address (*Address2*)
- City name of address (*City*)
- State name of address (*State*)
- Zip code of address (*Zip*)
- Customer telephone number (*Phone*).

A segment of List\_B is shown in Fig. 3.

```
"RecID","Name","Address","CityStateZip","POBox","POCityStateZip","SSN","DOB"
"A953698","antonio v cardona","247H HAHN ST","San Francisco, Cali 94134","PO BOX
280911","SAN FRANCISCO, CA 94128",196-36-9947,""
"A989582","ANTONIO V CARDONA","5221 ZELZAH AVEN APT219","encin, california
91316","PO BOX V19412","encino, ca 91416",196369974,"1913"
```

**Fig. 2** A segment of List\_A

RecID	FirstName	LastName	StrNbr	Address 1	Address 2	City	State	Zip	Phone
B932797	ANTONIO V	CARDONA	19412	APTDO		encino	ca	91416	818-453.1558
B949439	ANTONIO V	CARDONA	1207	Milj Way		stockton	ca	95209	(209)318-1443

**Fig. 3** A segment of List\_B

C967431	ANTONIO	V	CARDONA	196-36-9974	1913
(818)453.1558					

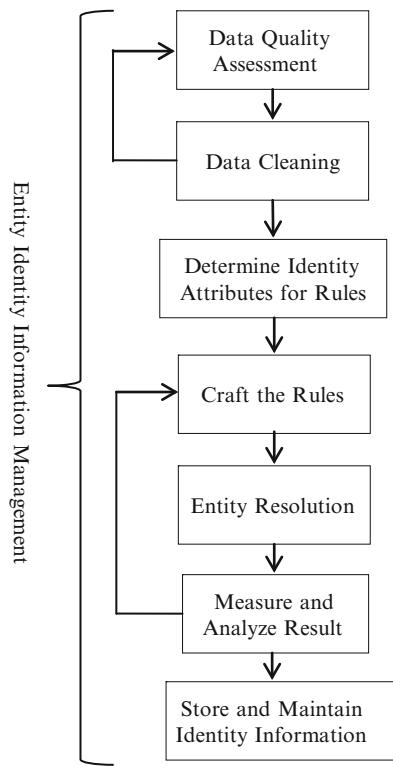
**Fig. 4** A segment of List\_C

The List\_C contains 76,059 references in a fixed-length field format. The attributes in List\_C have been sent to ABC by a data provider who did not provide a file layout. It has been left to the IT employees of ABC to decide where each field starts and ends and the content of each field. A segment of List\_C is shown in Fig. 4.

### 3.2 Overview of the Process

Figure 5 gives an overview of the entire process. The process starts with a data quality assessment of each source. The assessment includes profiling the sources and also visually inspecting the records to identify whether attributes can be used as identity attributes and, if so, to assess what standardization or other transformations and cleaning processes may need to be applied. For example, it is obvious from just looking at the file segments in Figs. 2 and 3 that the name attributes in List\_A and List\_B appear to combine some of the name elements that are handled separately in List\_C in Fig. 4, and the telephone numbers within and across List\_B and List\_C have different formats.

Based on the findings from the preliminary data quality assessment, the next step is to decide on exactly which data standardization and cleaning steps are needed. A second profiling is suggested after the data cleaning process to make sure the cleaning is effective and to assist in selection of the primary and supporting identity attributes. Primary identity attributes are those attributes which singly or in combination with other attributes are the most complete and have the highest level of uniqueness. For example, with student data a school-assigned student number would by itself probably meet these requirements. It would likely be present on all enrollment records and would have a high degree of uniqueness, though it may not

**Fig. 5** ER process overview

be 100 % unique as it may change as a student moves from school to school or from grade to grade. On the other hand, a student first name field and last name field would best qualify as primary identity attributes in combination rather than separately.

Supporting identity attributes are those attributes that tend to have less completeness or lower levels of uniqueness than primary identity attributes. For example, in student data, gender of male or female may be very complete but has a very low level of uniqueness. However, it could act to confirm or deny identity decisions based on primary attributes. For example, suppose that two student records both have the last name value of JOHNSON, but the first name value in one record is C and in the other it is CHARLES. If both records have a gender value of MALE, it would provide support along with the name values these may be records for the same student. On the other hand, if the records have different gender values, it would likely overrule any decision that these records are for the same student.

After the selection of primary and supporting attributes, the next step is to craft the first set of identity rules. The OYSTER system follows the Fellegi-Sunter model for record linking [8] in which pairs of records are judged to be “link” or “non-link” pairs (or perhaps “possible link” pairs) depending upon which pairs of attribute values agree or disagree. For example, the pattern that two student enrollment

records agree on first name, last name, and date-of-birth values, but disagree on school identifier values, might be designated a link rule, i.e., the decision is that the records are for the same student. Matching rules that give a yes or no (true or false) decision are called Boolean matching rules [77, 78]. There is also another family of entity resolution rules that compute a numerical score representing the distance between the records.

Once the identity rules have been crafted, the next step is to run the actual resolution process using an ER application, such as OYSTER. The result of an ER process is a grouping or partitioning of the records being resolved. As the ER system applies the identity rules to decide which input records represent the same identity, it gives those records a common identifier or label called *link*. It is for this reason that an ER process is sometimes called *record linking*. The groups of records that share the same link value are called a *cluster*. The resulting clusters coming out of the ER process have to be measured and analyzed in order to understand the performance of the identity rules. Based on the analysis of the cluster results, new rules may have to be added or existing rules may have to be modified or removed. Therefore, the steps from crafting the rules to measuring and analyzing the result comprise an iterative process.

The final step is to store the identity information and maintain so that it can be reused in later processes. Often one of the objectives is to be able to consistently assign records representing the same identity with the same label over time. These *persistent entity identifiers* are particularly important for master data management (MDM). The following sections will discuss each of the steps involved in detail by using the files described in Figs. 2–4.

### 3.3 Data Quality Assessment and Data Cleaning

Data quality assessment usually is critical, but unfortunately it is an often overlooked first step in any data-driven process [6]. A simple visual inspection of the data can provide a wealth of information. For example, for the ABC list data it is obvious that the three sources are in text files and in different formats. List\_A is in *comma-delimited* format (sometimes called *comma-separated values* or CSV format). It also uses quotation marks as a *text qualifier*. Text qualifiers are often used in delimited format files to avoid confusion about the delimiter value. For example, a comma is often used as a field separator, but it could also be used as part of a field value. If a quotation mark is used as a text qualifier, it means that any characters enclosed by quotation marks are part of the field value and are not considered as field separators. List\_B has a pipe-delimited format without a text qualifier. List\_C is in fixed-length field format and does not provide a field layout.

Another observation is that the attributes are not uniform across the sources. For example, List\_A has an attribute for the full name, but List\_B has attributes for first name and last name. There are also some other obvious data quality issues such as the different punctuation of telephone numbers and social security numbers.

Most data profiling tools have the ability to perform a *pattern frequency analysis*. In a pattern analysis individual characters are replaced by a token character to emphasize the character patterns. A typical convention is to replace any digit with the character “9,” any uppercase letter with an “A,” any lowercase letter with an “a,” and to leave all other characters in place. Under this convention, the value “818–453.1558” in record B932797 of Fig. 3 would produce the pattern “999–999.9999,” whereas the value “(209)318–1443” in record B949439 of Fig. 3 would produce the pattern “(999)999–999.” The pattern frequency analysis will provide the count of all values that follow that same pattern, making it valuable tool for determining if all values are consistently represented, and if not, what other patterns that have been used.

After observing these and other data quality condition in the ABC customer lists, the following steps are taken for this demonstration:

Step 1: Load the three files into a MySQL database.

The reason for loading the data into MySQL is because it is easier to manipulate the data in a database than in a text file: SQL queries can be constructed flexibly to check the data.

Step 2: Make a best guess at which attributes are in List\_C and the starting and ending position of each one.

Even though the List\_C did not come with a file layout, it can be inferred from the data profile reports and through observation of the values and patterns. For example, every value in the first 7 columns has the pattern “C999999,” making it a safe assumption that this represents the unique record identifier. Similar assumptions can be made on other attributes. The final attribute names for List\_C are:

- Unique record identifier (RecID)
- Customer first name (FirstName)
- Customer middle name (MiddleName)
- Customer last name (LastName)
- Customer Social Security Number (SSN)
- Customer date of birth (DOB)
- Customer telephone number (Phone)

With the identification of the attributes in List\_C, a preliminary master list of attributes can be built to compare the commonalities among the three lists. Table 1 shows the union of the attributes that have been identified in the three sources.

The next step is to parse and consolidate the attributes to be uniform across sources.

Step 3: Table 1 exposes a common problem in data management. Even though each of these lists would be considered a structured data source, it is evident from Table 1 that they are not entirely compatible with one another.

Across the sources there are some attributes which are not uniform. For example, Name in List\_A represents the entire name, whereas the First Name and Last Name in List\_B only represent parts of the name. An identity rule trying to match Name to First Name would not succeed. In this example Name is actually an unstructured

**Table 1** The original master attributes list

Attribute	List_A	List_B	List_C
RecID	X	X	X
Name	X		
FirstName		X	X
MiddleName			X
LastName		X	X
Address	X		
StrNbr		X	
Address1		X	
Address2		X	
CityStateZip	X		
City		X	
State		X	
Zip		X	
POBox	X		
POBoxCityStateZip	X		
SSN	X		X
DOB	X		X
Phone		X	X

field with respect to First Name and Last Name. Extracting elements of a single field into its component parts is a data operation called *parsing*.

The parsing strategy has to take into account both semantic and syntactic considerations. First, the parsed attributes used in identity rules need to correspond semantically (same meaning) to attributes in other sources. Another consideration is syntax or format. For US addresses, two fields for representing the state component of the address may have the same meaning, but may have different syntax. For example, one may have the name of the state spelled out as “Arkansas,” whereas the other may use the USPS standard two-letter code “AR.”

In the example given here, the fields were deliberately given different labels, Name vs. First Name and Last Name, to make the distinction apparent. However, this is not always the case and one should always beware that even though two fields are given the same label, it does not always mean they have the same meaning or syntax. *Data standards* are rules for consistently labeling data items so that two fields in different datasets that have the same label will have the same semantic and syntactic content. However, even when data standards are observed within an organization, there can still be a naming collision when data is brought in from outside of the organization. Many data cleaning tools provide parsing capabilities. In the example given here, the parsing was done using SAS DataFlux dfPower Studio® 8.2 [45], and the parsing scheme is shown in Table 2.

After parsing the new master list of attributes is shown in Table 3. Compared with the master attributes list in Table 1, it is now more fine grained and uniform across the three source lists.

Step 4: The next step is to apply some basic data standardization transformations before attempting the entity resolution process. For this example,

**Table 2** Attribute parsing scheme

Source	Original attribute	Parsed attributes
List_A	Name	First Name
		Last Name
List_A	Address	Pre-direction
		Street Number
		Street Name
List_A	City State Zip	City
		State
		Zip
List_A	POCity State Zip	PO City
		PO State
		PO Zip
List_B	Address1	Pre-direction Street Name

**Table 3** Master list of attributes after parsing

Attribute	List_A	List_B	List_C
RecID	X	X	X
FirstName	X	X	X
MiddleName			X
LastName	X	X	X
PreDirection	X		
StrNbr	X	X	
StrName	X	X	
Address2		X	
City	X	X	
State	X	X	
Zip	X	X	
POBox	X		
POCity	X		
POState	X		
POZip	X		
SSN	X		X
DOB	X		X
Phone		X	X

the standardizations are shown in Table 4 and were again done using a combination of DataFlux dfPower Studio® 8.2 and some simple Java programs.

Step 5: After standardization, the sources should be in a better shape for the entity resolution process. Now it is a good time to set a quality checkpoint and do a second run data profiling. The second check serves two purposes: to ensure that the transformations were successful and that the cleaning process itself did not create new problems. Interestingly, one of the sources of data quality errors in a system is data cleaning [4]. There are many places where things can go wrong such as errors in programming, incorrect job setup, unanticipated characters or patterns in the data, and many other issues that can cause unanticipated results. Reassessment through visual inspection and data profiling including value and pattern frequency reports

**Table 4** Standardization operations for some attributes

Attribute	Standardization operations
First Name	• Change to all uppercase
Last Name	• Remove all non-letter characters
Middle Name	
SSN	Remove all non-digit characters
Phone	
PO Box	Extract the PO box number (Java program)
State	Standardize to two-letter USPS state code
PO State	

is always a good idea. The profiling reports produced below are generated by SAS DataFlux dfPower Studio® 8.2.

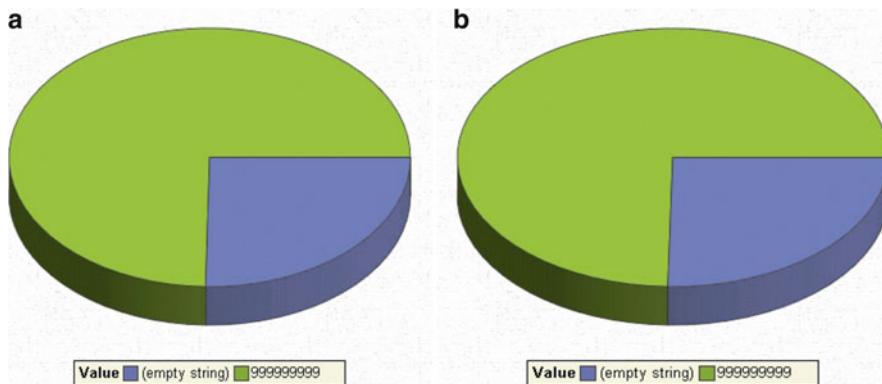
For example, Fig. 6a, b shows the pattern distribution in a pie chart format for the standardized SSN values in List\_A and List\_C, respectively. After standardization, both SSNs only have one pattern of nine consecutive digits (999999999).

### 3.4 Selecting Identity Attributes and Crafting the Identity Rules

The first step in crafting the identity rules is to select the attributes from the ABC customer data to serve as primary identity attributes and those that are best used as supporting identity attributes. Primary attributes are those attributes which singly or in combination with other attributes are the most complete and common across sources and also have a high level of uniqueness. The *SSN* attribute would ordinarily be an ideal primary attribute because it is expected to be unique for each individual, but the problem here is that it is only found in List\_A and List\_C, and it does not have a high level of completeness. For example, the data profiles of List\_A and List\_C show that the *SSN* attribute is only 74.68 % and 74.63 % complete, respectively.

A better choice for primary identity attributes in the ABC customer data are the first and last name items (*FirstName* and *LastName*). Both items are very complete as shown in Table 5 and, as Table 3 shows, they are the only attributes common to all three lists. Although there are many customers with the same first name and also many with same last name, the combination of first and last name taken together does provide a relatively high level of uniqueness. At the same time, the combination is not 100 % unique because it is easy to find instances where different customers have the same first and last names. The role of the supporting identity attributes is to help disambiguate these cases. For example, if two customer records have the same first and last name and also both have the same *SSN* value, then this would be a very strong confirmation that they are referring to the same customer. This set of conditions then becomes the first identity rule.

Rule 1: *FirstName* values are the same, the *LastName* values are the same, and the *SSN* values the same.



**Fig. 6** (a) List\_A SSN pattern distribution and (b) List\_C SSN pattern distribution

**Table 5** Completeness report for List A, B, and C

Attribute	List	Completeness
FirstName (Standardized)	A	98.2 %
FirstName (Standardized)	B	98.3 %
FirstName (Standardized)	C	93.3 %
LastName (Standardized)	A	100 %
LastName (Standardized)	B	100 %
LastName (Standardized)	C	100 %

In addition to selecting primary and supporting identity attributes, another factor in the design of identity rules are the choices for similarity functions. Even after extensive data cleansing and standardization, there may still be values that are not exactly the same but are similar enough to be considered a match. Some of the choices for general-purpose similarity functions were discussed in the previous section.

More specialized similarity functions can also be designed to detect a specific issue. One example is the Transpose function. It is a Boolean function returning the value of True if the two strings differ by exactly one transposition (reversal) of two consecutive characters and False otherwise. Another is to compare only specific portions of a string. For example, the rightmost substring function (SubStrRight) is a Boolean function that takes an integer parameter N. It returns the value True when the rightmost N characters of the two strings are exactly the same.

For example “JOANNE” and “LUANNE” would return True for the similarity function SubStrRight(4) because they agree on the last 4 characters.

Five additional identity rules that employ some of the similarity functions discussed above can be crafted following the same logic of primary attributes confirmed by supporting attributes. These five rules are as follows:

Rule 2: *FirstName* and *Lastname* values have a Levenshtein rating of 0.80 or more, and the SSN values differ by one transposition of consecutive digits.

Rule 3: *FirstName* and *Lastname* values have a Levenshtein rating of 0.80 or more, and the StrNbr values are the same.

Rule 4: *Lastname* values have a Levenshtein rating of 0.80 or more, and the last 6 digits of the Phone values are the same.

Rule 5: *FirstName* values are Nicknames (from nickname list), *Lastname* values have a Levenshtein rating of 0.80 or more, and the SSN values differ by one transposition of consecutive digits.

Rule 6: *FirstName* values are Nicknames (from nickname list), *Lastname* values have a Levenshtein rating of 0.80 or more, and the StrNbr values are the same.

The run configuration of the OYSTER ER system is established through a series of XML scripts called the *Run Script*, *Attributes Script*, and the *Source Descriptors*. The Attributes Script has two sections. The first section specifies the labels of the identity attributes, and the second section defines the identity rules. Each identity rule is given as a group of *<Term>* elements. Each *<Term>* element specifies which attribute values are to be compared (Item) and which similarity function is to be used for the comparison (MatchResult). Figure 7 shows the five ABC Company identity attributes and the six ABC Company identity rules as they would be defined in an OYSTER Attributes Script. The MatchCode value “LED(0.80)” is used to invoke the normalized Levenshtein edit distance function with a similarity threshold value of 0.80.

The form of the identity rules described here are simple Boolean matching rules where there is an implied logical AND between the terms in a rule and an implied logical OR between rules. In other words a single rule will be True (satisfied) only when every one of the terms in the rule has the degree of similarity specified. For example, in Rule 3, both the *FirstName* and *Lastname* values must have at least a 0.80 rating, and the *StrNbr* values must be exactly the same in order for the rule to be True. Otherwise, if any one of these conditions fails, the rule is False. Note that in OYSTER missing or null values are not used by the similarity functions. So even if both *StrNbr* values are null or blank, they will not be considered an exact match, i.e., the rule evaluation will be false.

At the same time the OR logic between rules means that two records only have to satisfy one of the identity rules to be considered equivalent. Thus, adding new rules can potentially increase the number of matching records, although this number may actually stay the same if new rules do not find any matches that already found by prior rules.

There are many other types of identity rules. Three other types are cross-attribute comparison rules, conflict rules, and distance rules. The cross-attribute rules allow

**Fig. 7** Attributes script for ABC lists

```

<?xml version="1.0" encoding="UTF-8"?>
<OysterAttributes System=ABC">
    <Attribute Item="FirstName" Algo="none" />
    <Attribute Item="LastName" Algo="none" />
    <Attribute Item="SSN" Algo="none" />
    <Attribute Item="StrNbr" Algo="none" />
    <Attribute Item="Phone" Algo="none" />
    <IdentityRules>
        <Rule Ident="1">
            <Term Item="FirstName" MatchResult="Exact"/>
            <Term Item="LastName" MatchResult="Exact"/>
            <Term Item="SSN" MatchResult="Exact"/>
        </Rule>
        <Rule Ident="2">
            <Term Item="FirstName" MatchResult="LED(0.80)"/>
            <Term Item="LastName" MatchResult="LED(0.80)"/>
            <Term Item="SSN" MatchResult="Transpose"/>
        </Rule>
        <Rule Ident="3">
            <Term Item="FirstName" MatchResult="LED(0.80)"/>
            <Term Item="LastName" MatchResult="LED(0.80)"/>
            <Term Item="StrNbr" MatchResult="Exact"/>
        </Rule>
        <Rule Ident="4">
            <Term Item="LastName" MatchResult="LED(0.80)"/>
            <Term Item="Phone" MatchResult="SubStrRight(6)"/>
        </Rule>
        <Rule Ident="5">
            <Term Item="FirstName" MatchResult="Nickname"/>
            <Term Item="LastName" MatchResult="LED(0.80)"/>
            <Term Item="SSN" MatchResult="Transpose"/>
        </Rule>
        <Rule Ident="6">
            <Term Item="FirstName" MatchResult="Nickname"/>
            <Term Item="LastName" MatchResult="LED(0.80)"/>
            <Term Item="StrNbr" MatchResult="Exact"/>
        </Rule>
    </IdentityRules>
</OysterAttributes>

```

for values of different attributes to be compared. Cross-attribute comparison rules address the issue of *misfielding* [79], a condition that occurs when a value intended for one attribute is assigned to a different attribute. This often occurs in names when one source of information records names in first-name-first format and another in last-name-first format. When the two sources are combined, it often results in all of the names in one of the sources to be misfielded where the first names are in the last name field and vice versa. In some cases it comes from a lack of clarity in requesting information such as primary and secondary addresses or telephone numbers. Cross-attribute comparison rules allow the user to design rules that compare first name to last name or primary telephone to secondary telephone to allow matches when this kind of information has been misfielded.

On the other hand, conflict rules are a sort of anti-rule to overcome the problems that sometimes come because of *transitive closure* [80]. A desirable characteristic of an ER process is that it should be *sequence neutral* [81], which means that it should give the same results regardless of the order in which the input records are processed. To insure that this happens, an ER system must observe *transitive equivalence* [6]. Transitive equivalence means that if the process decides that Record A is equivalent to Record B by one of the identity rules, and if the same process decides that Record B is equivalent to Record C by one of the identity rules, then the process must also treat Record A as equivalent to Record C even if A and C do not match by any of the identity rules. In other words, even though matching is not transitive, equivalence must be transitive, i.e., if A matches B and B matches C, then A does not necessarily match C, but A must be considered equivalent to C.

It is because of this transitive closure rule that when a record is found to match two different clusters of records, the two separate clusters should be merged into a single cluster. However, this can sometimes be a potential problem because several different identity rules may have been used to form each of these clusters. For example, suppose that Record A has an *Age* attribute with a non-null value, but Record A matches to Record B by a rule that does not consider the *Age* attribute because the *Age* value in Record B is missing. Similarly, suppose that Record C has a non-null *Age* value and it also matches Record B by a rule that does not consider *Age*. By transitive equivalence, Records A, B, and C should all be considered equivalent. However, further suppose that the *Age* value in Record A is significantly different than the *Age* value in Record C. This represents a conflict or inconsistency and indicates that even though Record B matches both Record A and Record C, the conflict in *Age* values should override the identity rules and the records should not be merged into a single cluster. A conflict rule prevents two clusters of records from merging if the values of a certain attribute or attributes are inconsistent between the two clusters.

Yet another type of rule is a distance rule. Where a Boolean rule always gives a True/False result from a comparison, a distance rule computes a numerical value that represents the distance between two records. A distance rule essentially becomes a Boolean rule whenever a threshold is established to decide when the distance measure is small enough that the two records are considered to be a match.

### 3.5 Entity Resolution Process

Once the reference sources have been analyzed, and a first set of identity rules has been crafted, the next step is to run an ER process that will systematically apply the rules to pairs of records in the reference source and create the identity clusters. The systematic comparison of the references is called an ER algorithm. There are many types of ER algorithms, and the choice of the algorithm depends upon several factors. Perhaps the most important consideration is whether the identity rules will be required to perform *record-based matching* or *attribute-based matching*.

In a record-based matching scheme, the values compared by identity rules must all come from the same record, whereas in attribute-based matching, the values can come from different, but equivalent, records.

The difference between record-based and attribute-based matching can be understood through a simple example based on student enrollment records in a school shown in Table 6. Each record in Table 6 has four attributes, a unique reference identifier (RefNbr), the student's first name (First), the student's last name (Last), and the identifier assigned to a student by one of the schools in the system that he or she attended (SID).

Table 7 defines two identity rules to be used in the ER algorithm. The first rule states that the ER process will consider two references (or EIS) to be equivalent if they have exactly the same first and last name. The second rule states that they are considered equivalent if they have the same school identifier regardless of the name values.

Since there are only 3 records, the algorithm can simply evaluate every possible pair-wise comparison. Record 1 compared to Record 2 is not a match by either Rule 1 or Rule 2. Similarly, Record 1 compared to Record 3 is not a match by either rule. However, it is true that Record 2 and Record 3 will match according to Rule 2 because they both have the same SID value. So at this point there are two clusters of equivalent records, Cluster 1 just containing Record 1 by itself and Cluster 2 containing both Record 2 and Record 3.

It is at this point when there are clusters of more than one record that the difference between record-based and attribute-based matching becomes important. The last possible comparison is to compare Cluster 1 to Cluster 2. Under a record-based matching constraint, the identity rules must compare values that come from the same record; therefore, comparing Cluster 1 to Cluster 2 is the same as comparing Record 1 to Record 2 and comparing Record 1 to Record 3. As was already seen, neither of these comparisons are a match. Since there are no more comparisons to be made, the process ends with two clusters.

$$\text{Record-Based ER Result} = \{\text{Cluster 1, Cluster 2}\} = \{\{\text{Record 1}\}, \{\text{Record 2, Record 3}\}\}$$

However, with attribute-based matching the result is different. In attribute-based matching, the identity rules can use any attribute value within the same cluster. When Cluster 1 is compared to Cluster 2, Rule 1 will be satisfied by selecting "John" as the value for First from Record 2 and selecting "Doe" as the value for Last from Record 3. Therefore, by Identity Rule 2, Cluster 1 matches Cluster 2 and should be merged into a single Cluster 3 containing all three records. Again since no more comparisons are possible, the process ends with a single cluster.

$$\text{Attribute-Based ER Result} = \{\text{Cluster3}\} = \{\{\text{Record 1, Record2, Record3}\}\}$$

As long as the ER process employs record-based matching, it will be sufficient to compare each record to every other record only one time. The simplest form of this algorithm is called the one-pass algorithm [71]. It starts with the input list of records

**Table 6** School enrollment records

RefNbr	First	Last	SID
1	John	Doe	G45
2	John	Doel	H37
3	Jon	Doe	H37

**Table 7** Identity rules

Rule	First	Last	SID
1	Exact	Exact	*
2	*	*	Exact

to be processed and an output list that is empty. The first input record is moved to the output list and becomes a one-record cluster. The second input record is compared to the first input record that is now in the output list. If it matches it is added to the cluster, otherwise it forms a new cluster. Next the third input record is compared to the two records in the output list. If it matches one of them, then it is added to the appropriate cluster. If it matches both and the first two records formed different clusters, then by transitive closure, all three records must form a single cluster. Input records that cause different cluster to merge in this way are sometimes called *glue records*. If the third record does not match either of the first two records, then it forms a new cluster.

This process continues by successively processing each input record and matching it against the records in the output list. Depending on how it matches, the input record may merge with an existing cluster, cause several clusters to combine, or form a new cluster. In any case the process is complete when the last input record is processed. If there are  $N$  records in the input, then the worst case number of comparisons required will be

$$\frac{N \cdot (N - 1)}{2}$$

For the sake of efficiency, very few systems fully execute the one-pass ER algorithm. This is especially true if some of the identity rules require exact matching. For example, if an identity rule required an exact match on a student's last name, then there would be no point in comparing an input record to every other record already processed because it only needs to be compared to other records that have the same last name value. There are several ways of organizing records into groups based on matching characteristics. Three of these techniques are *blocking*, *indexing*, and *sorted match keys*.

In blocking the records are divided into groups based on common values of one or more identity attributes. For example, when identity rules are matching on US address attributes, the records might be divided into blocks based on the postal zip code of the address. When an input record is processed, it would only be compared to other records within the same zip code block. Some blocking methods employ strategies to create blocks based on more than one attribute as a way to overcome the problem that any one attribute value may be missing or incorrect.

Indexing is similar to blocking but more dynamic. As an input record is processed the values of some or all of its identity attributes are inserted into an inverted index. At the same time these values are used to find previously processed records that

share some of the same values. For example, if an input school enrollment record has a student last name value of “JONES,” then all previously processed records that also had a last name value of “JONES” could be retrieved using the inverted index. Previously processed records retrieved in this way, as possible matches to an input record, are called *match candidates*.

Another strategy for reducing the total number of comparisons is to use sorted match keys. Very similar to blocking and indexing, this method relies on the creation of a hash key by extracting parts of identity attribute values and concatenating them into a single string. For example, the hashing algorithm might take the first letter of the first name, the first three consonant letters of the last name, and the first three digits of the postal zip code. The records are then sorted in hash key order with idea that for a given record, its potential match candidates will be near to it in the sorted list. The sorted hash key is often used with an ER algorithm called a *sliding window*. The window is the number of records on either side of a given record that is most likely to contain its match candidates. After all of the comparison have been made among the records in a given window, the center of the window moves (slides) down the list and the process is repeated.

In all of these methods there is a trade-off between efficiency and effectiveness. Whenever a decision is made to exclude certain records as match candidates, there is a probability that some matches will be missed. For example, in the sliding window algorithm the smaller the size of the window, the fewer comparisons that will have to be made within the window, but this increases the likelihood that a true match will lie outside of the window and not be found.

When an ER process uses attribute-based matching as opposed to record-based matching, the one pass and its variants will no longer be sufficient to find all of the possible matches with a given set of input records. As in the example, the problem is that when a new record is added to a cluster, its attribute values can combine with the attribute values of other records in the cluster that could potentially match with previously processed records.

The solution to this problem has been given through an ER algorithm called R-Swoosh [76]. The R-Swoosh algorithm proceeds in a similar manner to one pass except that when an input record is merged into an existing cluster in the output list, the new merged cluster must be placed back into the input list to be reprocessed. This is done in order to find any new matches that might have been created by adding the record to the cluster. Therefore, the input list may start with only single record to be processed, but as new clusters are formed and put back into the list, it will contain a mixture of records and clusters.

### 3.6 Measuring and Analyzing Results

From a data quality perspective, the ER/EIIM steps described here can also be thought of as following the total data quality management (TDQM) methodology [1]. TDQM is a continuous quality improvement process modeled on the manufacturing industry’s total quality management (TQM) [82] and adapted

for application to information systems. TDQM has four phases, namely, define, measure, analyze, and improve. The applicability of the methodology and its respective phases to ER/EIIM is outlined below:

- Define requirements: Because ER processes make decisions about the equivalence of references, ER errors are generally measured in terms of *false-positive* and *false-negative* rates. A false-positive error is the decision to link (cluster) two records that do not reference the same real-world entity. Conversely, a false-negative error is the decision not to link two records that are referring to the same real-world entity. For most practical applications it is almost impossible to eliminate these errors. The level of tolerance that is acceptable for these linking errors will depend upon the application for which the system is used. For example, in a customer identity management system, relatively high levels of false-negative errors are acceptable. From a customer relationship perspective, most businesses would consider it less of a problem to have mistakenly created two different accounts for the same customer than to have merged the accounts of two different customers. On the other hand, in a security-critical application most organizations would rather inconvenience many people than to let one bad actor go undetected, i.e., accept a high false-positive rate in order to attain a very low false-negative rate.
- Measure: Once the requirements or goals have been set, the next step is to measure the level of attainment. For ER processes this usually means measuring the false-positive and false-negative rates individually and, in instances, coming up with an overall accuracy through a single index measure. In either false-positive or false-negative case, the measurement must compare the linking provided by the process to the correct linking. Because, for most applications, the true linking for the entire set of records is not known, an estimate is made by taking these measures on a sample of the records.

Suppose that for some sample  $S$  of the records being processed the correct linking of these records has been determined. Then the correct linking of  $S$  divides it into clusters of equivalent records. These clusters form what is called a *partition* of  $S$ . A partition of  $S$  is a collection of nonempty subsets of  $S$  such that the union of all of the subsets is  $S$ , but no two of the subsets intersect each other. This has to be the case with clusters of equivalent records because every record must be in some cluster. At the same time, two clusters cannot overlap because, by transitive closure, any records in the intersection would have to be equivalent to all of the other records in both clusters, meaning there should not have been two clusters in the first place.

Let  $T$  represent the partition of  $S$  formed by the correct linking, i.e., the true clusters, and let  $X$  represent the partition of  $S$  formed by the linking from the ER process. If the process has made any false-positive or false-negative linking errors, then the partitions  $T$  and  $X$  will be different. In this case let  $V$  represent the partition formed by all of the nonempty intersections between the true clusters ( $T$ ) and the process clusters ( $X$ ). Then one measure of the accuracy of the process is given by the Talburt-Wang Index (TWI) defined by

$$\text{TWI} = \frac{\sqrt{|T| \cdot |X|}}{|V|}$$

The false-positive and false-negative rates can also be calculated based on the same sample, S. Let E represent the set of all pairs of records in the clusters of T, and let L represent the set of all pairs of records in the clusters of X. If  $\sim E$  represents all pairs of records in S that are not in E, and  $\sim L$  all pairs of records in S that are not in L, then the false-positive rate (FPR) and false-negative rate (FNR) of the process can be estimated by

$$\begin{aligned} \text{FPR} &= \frac{|L \cap \sim E|}{|\sim E|} \\ \text{FNR} &= \frac{|\sim L \cap E|}{|E|} \end{aligned}$$

- Analyze: The next step is to analyze the cause of the false-positive and false-negative errors caused by the identity rules. There are many reasons that they can occur such as poorly designed rules, too few identity attributes available, or because of data quality problems in the records. Adjusting the rules can be a delicate process because tightening (requiring a higher level of match) in one rule to prevent particular false-positive instances may have the unintended consequence of creating many new false-negative errors.
- Improve: After the error analysis, it is likely that some adjustment will need to be made either in the design of the rules or in the data quality preparation of the input records, or both.

Because TDQM is a quality process, these steps need to be repeated in order to gain continuous improvement. The goal is to iterate the TDQM process until a stable set of identity rules has been developed, identity rules that produce results that meet or exceed the requirements.

### ***3.7 Storing and Maintaining Identity Information***

All of the preceding discussion has centered on a one-time resolution of the records in a dataset, i.e., the ER phase. As with any type of information, entity identity information has a life cycle as new identities are created, updated, combined, and eventually discarded. EIIM systems have a continual influx of new reference information, and the resolution and integration of these references will impact the state of entity identity integrity of the system.

The key to sustaining entity identity integrity over time is the ability to save and reuse identity information gained in previous processes. In most record-linking

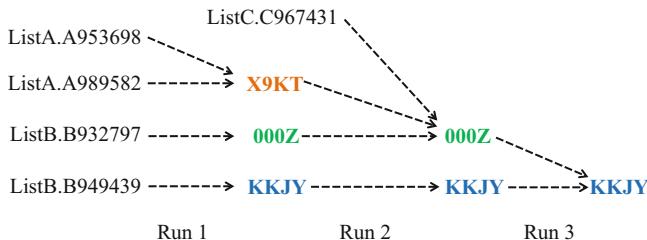
applications, once the effort has been expended to create the clusters of equivalent records, a single record called a *survivor record* is selected from the cluster for further processing, and the other records and the information they contain are discarded. If, instead, all of the information is brought forward, it allows future processes to reuse that effort and increase its knowledge about the identities being processed.

An EIIM process that builds and saves EISs created by a record-linking process is called an EIIM *identity capture configuration*. Note that each EIS is enclosed in <Identity> element of an XML document <root>. OYSTER assigns each EIS a unique 16-character identifier and a timestamp when the EIS is created. Within each EIS, the references which are brought in are stored with <Reference> elements. In order to save storage, the attributes' values are in a compressed, tagged format that is part of the Compressed Document Set Architecture (CoDoSA) [83]. The tags are defined in the <Attributes> section of the document. For example, “A” is the tag for the unique record identifier (RefID), “B” is the tag for the telephone number (Phone), and then the reference attribute's values in the <Value> tag is represented as “A^List B.B932797|B^8184531558|...”. In addition to <Value> element, there is another element <Traces> as child node of <Reference>. Traceability is a feature of OYSTER EIIM process. <Traces> element can have many <Trace> child nodes and each <Trace> element records one movement the corresponding reference takes. It keeps the OYSTER ID, Run ID, and Rule code with which the reference has been manipulated. This is important especially when the EIS gets updated or merged. Besides creating the EISs, OYSTER also produces a standard Link Index that simply shows the links assigned to each input record processed.

Beyond standard record-linking processes, EIIM systems also have the capability to add and modify EISs originally built in a previous process. This is an EIIM configuration called *identity update*. The new input references are added to the existing EISs by executing identity rules. And it may potentially cause the EISs to be merged. The story of how EISs are updated can be read in the <Traces> elements.

As discussed earlier, there is a limit to the accuracy of clustering obtained through the use of identity rules. Because these rules infer equivalence based on the information present in the records, they can only be as accurate as the information provided. Another type of resolution that is based on external knowledge that two records or two EIS are equivalent is called *asserted resolution*. Rather than inferring equivalence, the equivalence is known to be true. The OYSTER system supports four types of assertions, reference-to-reference assertion, reference-to-structure assertion, structure-to-structure assertion, and structure-split assertion. For example, if both EISs already existed, the form of assertion to be used is structure to structure, and the Rule code for this movement is “[@AssertStrToStr].”

Returning to the example of the ABC Company, Fig. 8 shows the process of how to maintain the identity information through 3 OYSTER runs. Run 1 is configured as *identity capture*. With identity rules in Fig. 7, it creates three EISs. The two references from List\_A formed identity “X9KT...” and the references from List\_B built identity “000Z...” and “KKJY...,” respectively. Run 2 is configured as *identity update*. The new coming source List\_C brought a new reference which



**Fig. 8** The identity information life cycle for the ABC company example

caused the two existing structures merged to one. The updated EIS “000Z...” now contains 4 references. A customer may self-report to ABC Company that he has moved and changed mailed addresses, thus making a connection between two different EISs for this customer. Run 3 is configured as *structure-to-structure assertion* to address this issue. Finally, EIS “KKJY...” is used to represent the most current view of this identity.

## 4 Summary

The ability to create and maintain persistent entity identity structures and identifiers over time is a critical process for master data management (MDM) and the support of entity-based information exchange systems, such as health information exchanges (HIEs). The aforementioned descriptions and examples only intended to provide a high-level overview of the EIIM process. For brevity, many process details have been omitted.

The good news is that for anyone interested in going deeper, there are several freely available tools to experiment with, including OYSTER ([sourceforge.net/projects/oysterer/](http://sourceforge.net/projects/oysterer/)), Talend® Open Studio for Data Quality (<http://www.talend.com/products/open-studio-dq.php>), and the synthetic data used for the examples ([ualr.edu/eriq](http://ualr.edu/eriq)).

## References

1. Lee Y, Pipino L, Funk J, Wang R (2006) Journey to data quality. MIT Press, Cambridge
2. English L (2009) Information quality applied. Wiley, Indianapolis
3. Dyché J, Levy E (2006) Customer data integration: Reaching a single version of the truth. Wiley, New York
4. Maydanchik A (2007) Data quality assessment. Technics Publications, Bradley Beach
5. Huang KT, Lee Y, Wang R (1999) Quality information and knowledge management. Prentice Hall PTR, Upper Saddle River
6. Talburt J (2011) Entity resolution and information quality. Morgan Kaufmann, Burlington
7. Lim E et al (1993) Entity identification in database integration. In: Proceedings of ninth international conference on data engineering, pp 294–301

8. Fellegi I, Sunter A (1969) A theory for record linkage. *J Am Stat Assoc* 64(328):1183–1210
9. Winkler W (1995) Matching and record linkage. In: Cox B et al (ed) *Business survey methods*. Wiley, New York, pp 355–384
10. Sarawagi S, Bhamidipaty A (2002) Interactive deduplication using active learning. In: Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining, pp 269–278
11. Neiling M, Jurk S (2003) The object identification framework. In: Proceedings of KDD03 workshop on data cleaning, record linkage, and object consolidation, pp 33–40
12. Mann G, Yarowsky D (2003) Unsupervised personal name disambiguation. In: Proceedings of the seventh conference on Natural language learning at HLT-NAACL, pp 33–40
13. Newcombe H (1967) Record linking: the design of efficient systems for linking records into individual and family histories. *Am J Hum Genet* 19(3):335–359
14. Newcombe H (1988) *Handbook of record linkage*. Oxford University Press, Oxford
15. Newcombe H, Kennedy J (1962) Record linkage: making maximum use of the discriminating power of identifying information. *Commun ACM* 5(11):563–566
16. Newcombe H et al (1950) Automatic linkage of vital records. *Science* 130(3381):954–959
17. Tepping B (1968) A model for optimum linkage of records. *J Am Stat Assoc* 63(324): 1321–1332
18. Herzog T, Scheuren F, Winkler W (2007) *Data quality and record linkage techniques*. Springer, New York
19. Hernandez M, Stolfo S (1998) Real-world data is dirty: data cleansing and the merge/purge problem. *Data Mining Knowl Discov* 2:9–37
20. Ananthakrishna R, Chaudhuri S, Ganti V (2002) Eliminating fuzzy duplicates in data warehouse. In: Proceedings of the 28th international conference on Very Large Data Bases (VLDB), pp 586–597
21. Wang R, Madnick S (1989) The inter-database instance identification problem in integrating autonomous systems. In: Proceedings of the fifth IEEE International Conference on Data Engineering
22. Cohen W, Kautz H, McAllester D (2000) Hardening soft information sources. In: Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining, pp 255–259
23. Bilenko M et al (2003) Adaptive name matching in information integration. *IEEE Intell Syst* 18(5):16–23
24. Elmagarmid A, Ipeirotis P, Verykios V (2007) Duplicate record detection: a survey. *IEEE Trans Knowl Data Eng* 19(1):1–16
25. Garcia-Molina H (2006) Pair-wise entity resolution: overview and challenges. In: Proceedings of the 15th ACM international conference on Information and knowledge management (CIKM '06), p 1
26. Berson A, Dubov L (2007) *Master data management and customer data integration for a global enterprise*. McGraw-Hill, New York
27. Bilenko M, Basu S, Sahami M (2005) Adaptive product normalization: using online learning for record linkage in comparison shopping. In: Proceeding of the fifth IEEE international conference on data mining (ICDM'05)
28. Quantim C et al (1998) How to ensure data security of an epidemiological follow-up: quality assessment of an anonymous record linkage procedure. *Int J Med Inform* 49(1):117–122
29. Hsiung P et al (2004) Alias detection in link data sets. In: Proceedings of the international conference on intelligence analysis
30. eHealth (2010) Key findings. In: eHealth Initiative. <http://www.ehealthinitiative.org/key-findings.html>. Accessed 9 Aug 2010
31. Inmon W, Nesavich A (2008) *Tapping into unstructured data*. Pearson Education, Crawfordsville
32. Freitag D (1998) Multi-strategy learning for information extraction. In: Shavlik J (ed) *Proceedings of the fifteenth international conference on machine learning*. Morgan Kaufmann, Burlington, pp 161–169

33. Cowie J, Wilks Y (1996) Information extraction. *Commun ACM* 39(1):80–91
34. Hashemi R et al (2002) Extraction of features with unstructured representation from HTML documents. In: Proceedings of international association for development of information society, pp 47–53
35. Bikell D, Schwartz R, Weischedel R (1999) An algorithm that learns what's in a name. *Mach Learn* 34:211–232
36. Liu B et al (2010) Refining information extraction rules using data provenance. *IEEE Data Eng Bull* 33:17–24
37. Blaschke C, Valencia A (2002) The frame-based module of the Suiseki information extraction system. *IEEE Intell Syst* 17:14–20
38. Agichtein E, Ganti V (2004) Mining reference tables for automatic text segmentation. In: Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining, pp 20–29
39. Califff M, Mooney R, Cohn D (2003) Bottom-up relational learning of pattern matching rules for information extraction. *J Mach Learn Res* 4:177–210
40. Soderland S (1997) Learning to extract text-based information from the World Wide Web. In: Proceedings of 3rd international conference on knowledge discovery and data mining, pp 251–254
41. Kimball R, Caserta J (2004) The data warehouse ETL toolkit: Practical techniques for extracting, cleaning, conforming, and delivering data. Wiley, New York
42. Chan Y, Talburt J, Talley T (2010) Data engineering: mining, information and intelligence. Springer, Norwell
43. Lindsey E (2008) Three-dimensional analysis, 1st edn. Data Profiling LLC
44. Borkar V, Deshmukh K, Sarawagi S (2000) Automatically extracting structure from free text addresses. *Bull Technical Committee Data Eng* 23:2000
45. SAS DataFlux (2011) Data management. <http://www.dataflux.com/Products/Data-Management-Studio.aspx>. Accessed 9 Mar 2011
46. Informatica (2011) Products. [http://www.informatica.com/products\\_services/Pages/index.aspx#page=page-8](http://www.informatica.com/products_services/Pages/index.aspx#page=page-8). Accessed 9 Mar 2011
47. IBM (2011) InfoSphere Platform. <http://www-01.ibm.com/software/data/identity-insight-solutions/>. Accessed 14 Sept 2012
48. Pushkarev V et al (2010) An overview of open source data quality tools. In Proceedings of information and knowledge engineering conference, Las Vegas
49. Talend (2012) www.talend.com. Accessed 14 Sept 2012
50. Ataccama (2012) DQ analyzer overview. <http://www.attaccama.com/en/products/dq-analyzer.html>. Accessed 14 Sept 2012
51. Pentaho (2012) Pentaho Kettle project. <http://kettle.pentaho.com/>. Accessed 14 Sept 2012
52. SQLPower Software (2012) Products. <http://www.sqlpower.ca/page/architect>. Accessed 14 Sept 2012
53. Naumann F, Herschel M (2010) An introduction to duplicate detection. Morgan & Claypool, San Rafael
54. Navarro G (2001) A guided tour to approximate string matching. *ACM Comput Surv* 33(1): 31–88
55. Levenshtein V (1965) Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR* 163(4):845–848
56. Smith T, Waterman M (2001) Identification of common molecular subsequences. *J Mol Biol* 147(1):195–197
57. Waterman M, Smith T, Beyer W (1976) Some biological sequence metrics. *Adv Math* 20(3):367–387
58. Sutinen E, Tarhio J (1995) On using q-gram locations in approximate string matching. In: Proceedings of 3rd annual European symposium on algorithms, pp 327–340
59. Holland G, Talburt J (2010) q-Gram Tetrahedral Ratio (qTR) for approximate string matching. In: Proceedings of 2010 Annual Axciom Laboratory for Applied Research Conference (ALAR-10)

60. Jaro M (1989) Advances in record-linkage methodology as applied to matching the 1985 Census of Tampa, Florida. *J Am Stat Assoc* 84(406):414–420
61. Porter E, Winkler W (1997) Approximate string comparison and its effect on an advanced record linkage system. In: Advanced Record Linkage System. U.S. Census Bureau, pp 190–199
62. Winkler W (1999) The state of record linkage and current research problems. Statistical Research Division, U.S. Census Bureau. Research Report
63. Budanitsky A, Hirst G (2006) Evaluating wordnet-based measures of lexical semantic relatedness. *Comput Linguist* 32(1):13–47
64. Russell R (1922) Patent No. US1435663 (A)
65. Knuth D (1989) Sorting and searching. *Art Comput Program* 3:391–392
66. Rajkovic P, Jankovic D (2007) Adaptation and application of Daitch-Mokotoff Soundex algorithm on Serbian names. In: Proceedings of XVII conference on applied mathematics, pp 193–204
67. Michelson M, Knoblock C (2006) Learning blocking schemes for record linkage. In: Proceedings of the 21st national conference on artificial intelligence, pp 440–445
68. Christen P (2012) Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection (data-centric systems and applications). Springer, New York
69. Hernandez M, Stolfo S (1995) The merge/purge problem for large databases. In: Proceedings of the ACM SIGMOD international conference on management of data (SIGMOD '95), 1995, pp 127–138
70. Yan S et al (2007) Adaptive sorted neighborhood methods for efficient record linkage. In: Proceedings of the 7th ACM/IEEE-CS joint conference on digital libraries, pp 185–194
71. Zhou Y, Talburt J (2011) Entity identity information management. In: Proceedings of the 16th International Conference on Information Quality (ICIQ-11), Adelaide, Australia, pp 327–341
72. Kalashnikov D, Mehrotra S (2006) Domain-independent data cleaning via analysis of entity-relationship graph. *ACM Trans Database Syst* 31(2):716–767
73. Nuray-Turan R, Kalashnikov D, Mehrotra S (2007) Self-tuning in graph-based reference disambiguation. In: Proceedings of the 12th international conference on database systems for advanced applications. Springer, Berlin, pp 325–336
74. Chen Z, Kalashnikov D, Mehrotra S (2007) Adaptive graphical approach to entity resolution. In: Proceedings of the 7th ACM/IEEE-CS joint conference on digital libraries (JCDL '07), pp 204–213
75. Xu X et al (2007) SCAN: a structural clustering algorithm for networks. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, pp 824–833
76. Benjelloun O et al (2006) Generic entity resolution in the SERF project. *IEEE Data Eng Bull*:13–20
77. Whang S, Garcia-Molina H (2010) Entity resolution with evolving rules. *Proc VLDB Endowment* 3(1–2):1326–1337
78. Zhou Y et al (2012) Implementing Boolean matching rules in an entity resolution system using XML scripts. In: Proceedings of the 2012 international conference on information and knowledge engineering (IKE'12), Las Vegas
79. Barateiro J, Galhardas H (2005) A survey of data quality tools. *Datenbank-Spektrum* 14:15–21
80. Li WN, Bheemavaram R, Zhang X (2009) Transitive closure of data records: application and computation. *Data Eng Int Ser Oper Res Manage Sci* 132:39–75
81. Jonas J (2006) Sequence neutrality in information systems. [http://jeffjonas.typepad.com/jeff\\_jonas/2006/01/sequence\\_neutra.html](http://jeffjonas.typepad.com/jeff_jonas/2006/01/sequence_neutra.html). Accessed 13 Sept 2012
82. Ahire SL (1997) Management science- total quality management interfaces: an integrative framework. *Interfaces* 27(6):91–105
83. Talburt J, Nelson E (2009) CoDoSA: a light-weight, XML framework for integrating unstructured textual information. In: 15th Americas conference on information systems. AIS Electronic Library, San Francisco, p 489

# Managing Quality of Probabilistic Databases

Reynold Cheng

**Abstract** Uncertain or imprecise data are pervasive in applications like location-based services, sensor monitoring, and data collection and integration. For these applications, *probabilistic databases* can be used to store uncertain data, and querying facilities are provided to yield answers with statistical confidence. Given that a limited amount of resources is available to “clean” the database (e.g., by probing some sensor data values to get their latest values), we address the problem of choosing the set of uncertain objects to be cleaned, in order to achieve the best improvement in the quality of query answers. For this purpose, we present the *PWS-quality* metric, which is a universal measure that quantifies the ambiguity of query answers under the *possible world semantics*. We study how PWS-quality can be efficiently evaluated for two major query classes: (1) queries that examine the satisfiability of tuples independent of other tuples (e.g., range queries) and (2) queries that require the knowledge of the relative ranking of the tuples (e.g., MAX queries). We then propose a polynomial-time solution to achieve an optimal improvement in PWS-quality. Other fast heuristics are also examined.

## 1 Introduction

Traditionally, a database assumes that the values of the data stored are exact or precise. In many emerging applications, however, the database is inherently uncertain. Consider a habitat monitoring system where data like temperature, humidity, and wind speed are acquired from sensors. Due to the imperfect nature of the sensing instruments, the data obtained are often contaminated with noises [17]. As another example, in the global-positioning system (GPS), the location values collected have some measurement error [30,36]. In biometric databases, the attribute

---

R. Cheng (✉)

Department of Computer Science, University of Hong Kong, Pokfulam Road, Hong Kong, China  
e-mail: [ckcheng@cs.hku.hk](mailto:ckcheng@cs.hku.hk)

values of the feature vectors stored are not exact [7]. Integration and record linkage tools also associate confidence values to the output tuples according to the quality of matching [15]. To deal with the increasing need of handling uncertainty, researchers have recently proposed to consider uncertainty as a “first-class citizen,” by managing data in an “uncertain database” [1, 4, 9, 15].

In these databases, queries can be evaluated to produce imprecise answers with probabilistic guarantees. The ambiguity of a query answer constitutes the notion of *query quality*, which describes “how good” a query answer is [9]. In this chapter, we address the issue of how to improve query quality, through the means of reducing the ambiguity of the database. Data imprecision can be alleviated in different ways. For example, in a sensor-monitoring application, a database system is used to store the current values of thousands of sensors deployed in a geographical region. Due to limited resources, the system may not be able to capture the sensor information at every point of time; instead, it uses the stored values to estimate the current sensor readings [9, 17]. To reduce the error of estimation, the system can “probe” a sensor, which responds to the system with its newest value. As another example, consider a database that captures the movie ratings, based on the fusion of the IMDB movie information and the user ratings obtained from the Netflix challenge. The database contains the customers’ ratings of each movie, represented as a probability distribution. The uncertainty about these ratings can be “sanitized” by contacting the respective customers for clarification. The resulting database, which is less uncertain than before, could then provide a higher-quality service.

Ideally, the whole database should be cleaned. In reality, this may not be feasible, since cleaning data can be costly. A sensor-monitoring system, for example, may only probe a small portion of sensors, partly due to the limited bandwidth in the wireless network and partly due to the scarce battery power of the sensing devices. As for the movie-rating database, it may be difficult to validate the ratings of all the customers who are involved in the movie evaluation. Generally, a cleaning operation is constrained, for instance, by a fixed “budget,” which describes the maximal amount of effort that can be invested for cleaning the data. The cleaning budget for a sensor-monitoring system can be the maximum amount of bandwidth that can be used for sensor probing. For the movie-rating database, such a budget can be the maximum number of man-hours allowed for verifying the movie ratings.

In this chapter, we address the problem of cleaning uncertain data for achieving better query or service quality, under a limited budget. Despite the importance of uncertain database cleaning, relatively little work has been done in the area (e.g., [2, 8, 17, 22, 28]). Our main idea is to make use of the query information to decide the set of data items to be cleaned. By operating on these data, the quality of the answers returned to the user can attain the highest improvement. Our solution is based on the *probabilistic database* [4, 15], a widely studied uncertain data model. The main challenges include the following:

- Define a sound and general quality metric over query results.
- Develop efficient methods to compute this metric.
- Devise efficient and optimal cleaning algorithms.

**Table 1** Uncertain database example

Product ID	Tuple ID	Price (\$)	Prob.
$a$	$a_1$	120	0.7
$a$	$a_2$	80	0.3
$b$	$b_1$	110	0.6
$b$	$b_2$	90	0.4
$c$	$c_1$	140	0.5
$c$	$c_2$	110	0.3
$c$	$c_3$	100	0.2
$d$	$d_1$	10	1

**Table 2** Results of the MAX query in Table 1

Tuple	Qualification probability
$a_1$	0.35
$b_1$	0.09
$c_1$	0.5
$c_2$	0.09
$c_3$	0.024

To illustrate, Table 1 shows a relation in a probabilistic database, which stores the quotations of four products (with IDs  $a$ ,  $b$ ,  $c$ , and  $d$ ), collected from Web pages by using some automatic schema-matching methods. An attribute called *existential probability* (Prob. in short) is used to indicate the confidence of the existence of each tuple. A tuple is also associated with an “x-tuple” [1], which represents a distribution of alternatives. For example, product  $a$  has a 0.7 chance for offering a price of \$120 and a 0.3 chance for having a quotation of \$80. Now consider the MAX query: “Return the tuple with the highest price.” Due to data imprecision, this query can produce imprecise answers. Table 2 shows the query result, which contains the IDs of the tuples, and their nonzero probabilities (or *qualification probabilities*) for being the correct answers. These queries, which produce answers with statistical guarantees, are generally known as *probabilistic queries* [1, 9, 15].

Based on the answer probabilities, a real-valued “quality score” can be defined to capture the degree of ambiguity of a query answer. For example, the score of the MAX query result (Table 2) is  $-1.73$  (according to our quality metric). Suppose the database in Table 1 is partially cleaned (e.g., by consulting the companies about the actual prices of the products). Table 3 shows one possible scenario, where the uncertainties associated with x-tuples  $a$  and  $c$  are removed. In this table, only one tuple exists for each of  $a$  and  $c$ , and the existential probability of this tuple is equal to one. The new result of the MAX query is shown in Table 4, with a lower ambiguity or an improved quality score of  $-0.97$ . In the extreme case, if all the x-tuples are cleaned, the quality score becomes the highest (a value of zero with our metric).

How should such a query quality metric be defined? In this chapter, we examine the *PWS-quality*, which was first proposed in [12]. This metric provides a *universal measure* of query quality (i.e., can be used by any queries) for the probabilistic database. It is essentially an entropy function [32], which returns a real-valued score for conveniently indicating the amount of imprecision in query answers.

**Table 3** A partially cleaned instance of Table 1

Product ID	Tuple ID	Price (\$)	Prob.
$a$	$a_2$	80	1
$b$	$b_1$	110	0.6
$b$	$b_2$	90	0.4
$c$	$c_3$	100	1
$d$	$d_1$	10	1

**Table 4** Results of the MAX query in Table 3

Tuple	Qualification probability
$b_1$	0.6
$c_3$	0.4

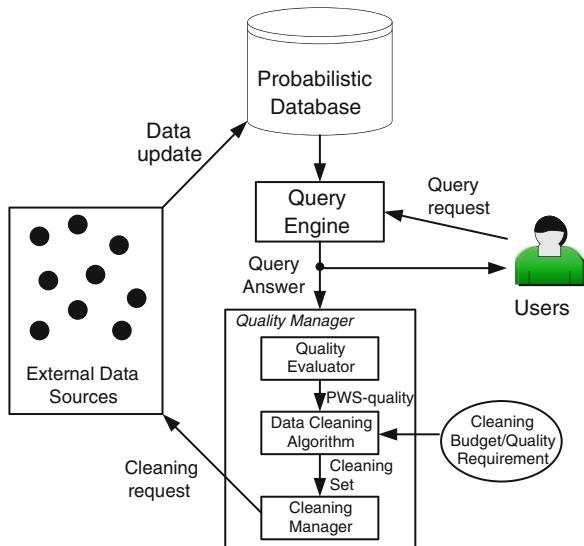
The PWS-quality also enables efficient data cleaning solutions, as we will show in this chapter.

Another salient feature of PWS-quality is that it assumes the *possible world semantics* (or PWS in short). The PWS provides a formal interpretation of the probabilistic database model [17], where a database is viewed as a set of deterministic database instances (called *possible worlds*), each of which contains a set of tuples extracted from each x-tuple. An example possible world for Table 1 contains the tuples  $\{a_1, b_2, c_3, d_1\}$ . Query evaluation algorithms for a probabilistic database should follow the notion of PWS, i.e., the results produced should be the same as if the query is evaluated on all the possible worlds [17]. Analogously, the PWS-quality score is calculated based on the query results obtained from all the possible worlds.

One apparent problem about the PWS-quality is that it is inefficient to calculate. This is because evaluating this measure requires examining all possible worlds, the number of which can be exponentially large [4, 15]. Interestingly, we observe that it is not often necessary to examine all the database instances; the PWS-quality can, in fact, be computed by using the query answers returned to the user. This is true for a broad class of queries known as the *entity-based* query [9]. This kind of query has the property that the final answer returned to the user contains the IDs of the tuples that satisfy it, as well as their qualification probabilities (e.g., Table 2). We study two representative examples of entity-based queries, namely, the range query and the MAX query. Both queries are used in many applications. For example, in a sensor-monitoring application, a range query can be as follows: “Return the IDs of the sensors whose temperature values are within  $[10^{\circ}\text{C}, 20^{\circ}\text{C}]$ .” In the movie database, a MAX query can be as follows: “Return the ID of the movie viewer whose rating is the highest.” We show that the PWS-quality of these two queries can be quickly computed by using query answer information. Our methods are effective because a query answer can be efficiently generated by existing query evaluation and indexing algorithms, and the complexity of our technique is linear to the size of the query answer.

The PWS-quality also serves as a useful tool for solving the data cleaning problem. Given the set of x-tuples to be cleaned, we prove that there is always a monotonic increase in the expected value of PWS-quality. This helps us to formulate

**Fig. 1** The framework of our solution



the data cleaning problem as follows: choose the subset  $X$  of  $x$ -tuples such that (1) the increase in the expected quality of cleaning the  $x$ -tuples in  $X$  is the highest and (2) the total cost of cleaning  $X$  does not exceed a given budget. This problem is challenging because calculating the expected quality improvement of  $X$  requires the processing of all combinations of the tuples in  $X$ . Moreover, a naïve approach of finding the optimal set  $X$  requires the testing of different combinations of  $x$ -tuples in the database, rendering an exponential time complexity. To solve these problems, we convert the PWS-quality expression into an “ $x$ -form”—a linear function of the probability information of the  $x$ -tuples. The  $x$ -form allows us to compute the expected quality improvement for cleaning a set of  $x$ -tuples easily. Moreover, it has the same format for both the range and the MAX queries (with different parameters), so that only a single solution is needed to support both queries. To find the optimal solution without testing all combinations of  $x$ -tuples from the whole database, we show that it is only necessary to select the  $x$ -tuples whose tuples appear in a query answer. We then model the cleaning task as an optimization problem and develop a dynamic-programming-based algorithm, in order to deduce the optimal set of  $x$ -tuples in polynomial time. We also propose other approximate heuristics (such as the greedy algorithm). Our algorithms serve both the range and the MAX queries. They also support databases that contain tuples with the same attribute value.

Figure 1 illustrates a system design that incorporates our solutions. The query engine, upon receiving one or more users’ requests, produces the answers of probabilistic queries. While the query answers are returned to the users, their information is also passed to the *quality manager*. Inside this module, the *quality evaluator* computes the PWS-quality score. It then sends the necessary information to the *data cleaning algorithm*, which derives the optimal set of  $x$ -tuples to be

cleaned (or “cleaning set”), with the available budget considered. The *cleaning manager* is responsible for performing the sanitization activity (e.g., requesting the selected sources to report their updated values). The queries, when executed again, will then have an improvement in their expected quality scores. Notice that the quality manager is decoupled from the query engine, since it only requires the probability information of the answer tuples. In this chapter, we focus on the design of the quality evaluator, as well as data cleaning algorithms.

The rest of this chapter is organized as follows: in Sect. 2 we present the related works. Section 3 discusses the data and query models. In Sect. 4 we present the formal notion of the PWS-quality and efficient methods for evaluating it. Section 5 describes the quality-based cleaning methods and other heuristics. Section 6 concludes.

## 2 Related Work

**Querying Uncertain Databases.** Due to its simplicity and clarity in semantics, the probabilistic database model [4, 15] has received plenty of attention. Particularly, the notion of  $x$ -tuples [1] has been commonly adopted as a formal model for representing the uncertainty of tuples. Dalvi et al. [15] demonstrated that evaluating queries using the notion of PWS can be inefficient, since an exponentially large number of possible worlds needs to be examined. Thus, researchers have proposed to modify the query semantics. For example, efficient solutions for different variants of top- $k$  queries are studied in [20, 31, 37, 40]. And in [24], a general function has been proposed to unify those semantics. Another well-studied data model is the “attribute uncertainty,” where attribute values are characterized by a range and a probability distribution function (pdf) [9, 36]. For this data model, efficient evaluation and indexing algorithms have been proposed, including range queries [39], nearest-neighbor [11, 23], MIN/MAX queries [9, 17], top- $k$  queries [38], skylines [29], and reverse skylines [25]. In [34], efficient query algorithms for uncertainty of categorical data are studied. Recently, a formal model for attribute uncertainty based on the PWS has been proposed [35]. In [5], the ULDB model is presented, which combines the properties of probabilistic and lineage databases. The Monte Carlo database system (DBMS) [3, 21] is proposed recently which used “VG function” to present uncertainty and apply Monte Carlo approach to process the queries. Although our work is based on probabilistic databases, the idea can potentially be extended to support other data models.

**Quality Metrics for Uncertain Data.** A number of quality measures have been studied. In [10, 17], if the qualification probability of a result is higher than a user-defined threshold, then the query result is considered to be satisfactory. In [33], the quality of a top- $k$  query is given by the fraction of the true top- $k$  values contained in the query results. In [9], different metrics are defined for range queries,

nearest-neighbor queries, AVG, and SUM queries. In these works, quality metrics are designed for specific query types. The PWS-quality, on the other hand, provides a general notion of quality that can be applied to any kind of queries. Thus, PWS-quality can provide a fair comparison of quality among the answers from different queries. Another quality metric, called the query reliability, is defined in [16, 19]. However, this metric was not studied in the context of probabilistic databases. Moreover, it is not clear how they can be applied to the problem of data cleaning.

**Cleaning Uncertain Data.** A number of works have studied the handling of ambiguous or inconsistent data. In sensor networks, efficient methods for probing fresh data from stream sources and sensor networks have been considered [8, 17, 26, 28]. A comprehensive survey in [18] summarizes the works done in *duplicate elimination*—the detection and deletion of similar and inaccurate information in a database. In [22], integrity constraints are used to clean dirty data. The authors in [2] examined the detection and merging of duplicate tuples in inconsistent databases. [6] discussed how to manage different versions of the resulting database, after removing duplicates from it. In [13], we studied the efficient removal of ambiguous data, but did not address the probabilistic database model. Complementary to these works, we study how the PWS-quality metric can be used to facilitate the cleaning of a probabilistic database.

### 3 Data and Query Models

We now describe the probabilistic data model in Sect. 3.1. We then discuss the types of queries studied in Sect. 3.2.

#### 3.1 The Probabilistic Database Model

A probabilistic database  $D$  contains  $m$  entities known as the “x-tuples” [1, 4, 17]. We denote the  $k$ th x-tuple by  $\tau_k$ , where  $k = 1, \dots, m$ . We also assume that the x-tuples are independent of each other. Each x-tuple is a set of tuples  $t_i$ , which represent a distribution of values within the x-tuple. There are a total of  $n$  tuples in  $D$ . Each tuple has four attributes:  $(ID_i, v_i, e_i, x_i)$ . Here,  $ID_i$  is a unique identifier of  $t_i$ , and  $v_i$  is a real-valued attribute used in queries (called the *querying attribute*). The attribute  $e_i$  is the existential probability of  $t_i$  – the probability that  $t_i$  exists in the real world. Each tuple belongs to one of the x-tuples, and  $x_i = \{k | k = 1, \dots, m\}$  denotes that  $t_i$  belongs to the  $k$ th x-tuple.

Within the same x-tuple, the existence of tuples is mutually exclusive. We also assume that the sum  $s_k$  of all  $e_i$ 's of the tuples in the same x-tuple  $\tau_k$  is equal to 1.<sup>1</sup> In Table 1, for example, there are four x-tuples  $(a, b, c, d)$ . The “Price” and “Prob.” columns represent the querying attribute and existential probability, respectively.

### 3.2 Queries

We study two types of entity-based queries: *non-rank-based* and *rank-based* [9]. In a non-rank-based query, a tuple's qualification probability is independent of the existence of other tuples. For example, queries whose selection clauses involve only the attributes of a single tuple belong to this query class. Another good example is the range query:

**Definition 1 (Probabilistic Range Query (PRQ)).** Given a closed interval  $[a, b]$ , where  $a, b \in \mathfrak{N}$  and  $a \leq b$ , a PRQ returns a set of tuples  $(t_i, p_i)$ , where  $p_i$ , the qualification probability of  $t_i$ , is the nonzero probability that  $v_i \in [a, b]$ .

For a rank-based query, a tuple's qualification probability is dependent on the existence of other tuples. Examples of this class include the MAX/MIN query and the nearest-neighbor query. We study the MAX query here.

**Definition 2 (Probabilistic Maximum Query (PMaxQ)).** A PMaxQ returns a set of tuples  $(t_i, p_i)$ , where  $p_i$ , the qualification probability of  $t_i$ , is the nonzero probability that  $v_i \geq v_j$ , where  $j \neq i \wedge j = 1, \dots, n$ .

Although the answers returned by both queries have the same form, their PWS-quality scores are computed in a different way, as illustrated in the next section. We will also discuss how PWS-quality can be computed for other entity-based queries (e.g., nearest-neighbor queries). Table 5 shows the symbols used in this chapter.

We now briefly explain how PRQ and PMaxQ can be evaluated efficiently (without consulting the possible worlds). A PRQ can be computed by examining each tuple  $t_i$  and testing whether its querying attribute,  $v_i$ , is within  $[a, b]$ . If this is not true, then  $t_i$ 's qualification probability,  $p_i$ , must be zero. Otherwise,  $p_i = e_i$ , its existential probability. The probability of  $t_i$  for satisfying the PMaxQ is the product of (1) its existential probability, and (2) the probability that x-tuples other than the one that  $t_i$  belongs to do not have a tuple with value larger than  $v_i$ . Indexing solutions (e.g., B-tree and R-tree) can be built on the querying attributes in order to improve the query performance.

Also, as mentioned in Sect. 1, a query may need to be reevaluated after cleaning is done. However, this round of query evaluation can be done more efficiently. In

---

<sup>1</sup>If  $s_k$  is less than 1, we conceptually augment a “null” tuple to  $\tau_k$ , whose querying attribute has a value equal to  $-\infty$  and existential probability equal to  $1 - s_k$ . This null tuple is only used for completeness in proofs; they do not exist physically.

**Table 5** Symbols used in this chapter

Notation	Description
<i>Data model</i>	
$D$	A probabilistic database
$\tau_k$	An x-tuple of $D$ , with $k = 1, \dots, m$
$t_i$	A tuple of $D$ with $i = 1, \dots, n$
$ID_i$	A unique identifier of $t_i$
$v_i$	Querying attribute of $t_i$
$e_i$	Existential probability of $t_i$
$x_i$	The ID of the x-tuple ( $k$ ) that contains $t_i$
<i>Query model</i>	
$Q$	A probabilistic query
$p_i$	Qualification prob. of $t_i$ for $Q$
$P_k$	Qualification prob. of $\tau_k$ for $Q$
<i>Quality metrics</i>	
$r_j$	A distinct PW-result ( $j = 1, \dots, d$ )
$q_j$	Prob. of occurrence of $r_j$
$S(D, Q)$	PWS-quality of $Q$ on database $D$
<i>Data cleaning</i>	
$C$	Cleaning budget for $Q$
$T$	Target quality improvement
$c_k$	Cost of cleaning $\tau_k$
$X$	Set of x-tuples to be cleaned
$I(X, D, Q)$	Quality improvement of cleaning $X$

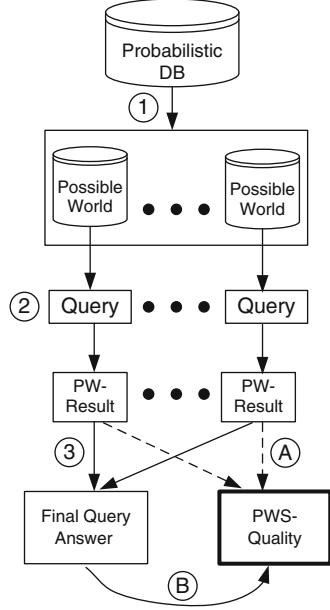
particular, only the x-tuples whose tuples appear in the query answer of the first evaluation round need to be considered. These details are discussed in [12].

## 4 The PWS-Quality

To understand this metric, let us first review how the possible world semantics (PWS) are used to evaluate a query. As shown in Fig. 2, a probabilistic database is expanded to a set of *possible worlds* (Step 1). The query is then issued on each possible world, producing answers that we call *PW-results* (Step 2). In Step 3, the PW-results are combined to produce the final query answer. For example, a possible world in Table 1 is the set  $W$  of tuples whose IDs are  $a_1$ ,  $b_2$ ,  $c_3$ , and  $d_1$ . If a MAX query is issued on `price`, the PW-result of  $W$  is  $a_1$  (since it has the largest `price`), with a probability of  $0.7 \times 0.4 \times 0.2 \times 1 = 0.056$ . In the final query answer, the qualification probability of  $a_1$  is equal to the sum of the probabilities that  $a_1$  is the answer in all the possible worlds, that is, 0.35.

The PWS-quality is essentially a function of the PW-results (computed in Step A in Fig. 2). Since the form of the queries for computing the PW-results is not specified, the PWS-quality can be applied to *any* type of queries. The major problem

**Fig. 2** PWS and PWS-quality



of this approach is that there can be an exponentially large number of possible worlds [4,15], so as the PW-results. Computing PWS-quality can thus be very costly. To address this problem, we show how PWS-quality can be evaluated by using the tuple information in the final query answers, instead of the PW-results (Step B), for PRQ and PMaxQ. Since the number of tuples in the query answer set is much smaller than that of the possible worlds, the computation efficiency of PWS-quality is also better.

Next, we examine the definition of PWS-quality (i.e., Step A), in Sect. 4.1. Then we propose a better method (i.e., Step B) in Sect. 4.2.

#### 4.1 Evaluating the PWS-Quality

The PWS-quality is essentially the entropy [32] of the PW-results produced in Step 2 of Fig. 2. Let  $\{r_1, \dots, r_d\}$  be the set of  $d$  distinct PW-results. Also, let  $q_j$  be the probability that  $r_j$  is the actual answer (we call  $q_j$  the *PW-result probability* of  $r_j$ ).

**Definition 3.** The *PWS-quality* of a query  $Q$  evaluated on a database  $D$ , denoted by  $S(D, Q)$ , is

$$S(D, Q) = \sum_{j=1}^d q_j \log q_j \quad (1)$$

Notice that the base of the  $\log()$  function is 2. Moreover, the sum of the PW-result probabilities must be equal to one (i.e.,  $\sum_{j=1}^d q_j = 1$ ). By using this fact and comparing with the entropy function [32], it can be shown that the PWS-quality [Eq. (1)] is the negated value of the entropy of the PW-results. The entropy, a popular function for measuring uncertainty in the information theory literature, is adopted here to quantify the impreciseness of query answers. The value of the PWS-quality score ranges from  $-\log d$  (i.e., the most ambiguous result) to zero (i.e., a single PW-result).

The problem of computing PWS-quality in this way is that we need to know all the PW-result probabilities. This may represent a performance bottleneck, since the number of PW-result possibilities, derived from possible worlds, can be exponentially large. This is true for a PRQ, where each PW-result contains a unique set of tuples whose querying attributes are inside the query range. For a PMaxQ, the number of PW-results can also be combinatorial, if more than one tuple contain the same querying attribute value. For instance, in Table 1, tuples  $b_1$  and  $c_2$  have the same `price` (i.e., \$110). We then have  $2^2 - 1 = 3$  PW-results that contain one or more of these tuples as the answer (i.e.,  $\{b_1\}, \{c_2\}$  and  $\{b_1, c_2\}$ ), derived from the possible worlds where all tuples with `price` above \$110 are excluded. Let us investigate how PWS-quality can be evaluated more efficiently for these queries.

## 4.2 The $x$ -Form of the PWS-Quality

The PWS-quality can in fact be computed by using the probability information of the tuples in the query answer (Step B of Fig. 2). In particular, the PWS-quality (for both PRQ and PMaxQ) can be converted to an expression known as the  $x$ -form. An  $x$ -form is essentially a sum of some function  $g$  evaluated on each  $x$ -tuple  $\tau_k$ , and  $g$  can be computed efficiently based on the probability information of the tuples in  $\tau_k$ . For notational convenience, we use  $Y(x)$  to denote the function  $x \log x$ . We also let  $P_k$  be the qualification probability of  $\tau_k$  (i.e., its probability for satisfying the query). Since tuples belonging to the  $x$ -tuple are mutually exclusive, we have

$$P_k = \sum_{t_i \in \tau_k} p_i \quad (2)$$

The following lemma presents an alternative formula of PWS-quality.

**Lemma 1.** *The  $x$ -form of the PWS-quality is given by*

$$S(D, Q) = \sum_{k=1}^m g(k, D, Q) \quad (3)$$

*For PRQ,*

$$g(k, D, Q) = \sum_{t_i \in \tau_k} p_i \log e_i + Y(1 - P_k) \quad (4)$$

For **PMaxQ**, let the  $i$ th tuple of  $\tau_k$  be  $t_{k,i}$ , sorted in descending order of  $v_{k,i}$ . If  $t_{k,i}$  has existential probability  $e_{k,i}$  and qualification probability  $p_{k,i}$ , then

$$g(k, D, Q) = \sum_{i=1}^{|t_k|} (p_{k,i} \log e_{k,i} + \omega_{k,i} \log(1 - \sum_{j=1}^i e_{k,j})) \quad (5)$$

where

$$\omega_{k,i} = \begin{cases} (1 - \sum_{j=1}^i e_{k,j}) \left( \frac{p_{k,i}}{e_{k,i}} - \frac{p_{k,i+1}}{e_{k,i+1}} \right) & i < |t_k| \\ 0 & i = |t_k| \end{cases} \quad (6)$$

Lemma 1 states that the PWS-quality is the sum of some function  $g$  for  $k = 1, \dots, m$ . Each  $g$  is a function of the existential and qualification probabilities of tuples within x-tuple  $\tau_k$ . The detailed proof of this lemma can be found in [12].

Interestingly, even though PRQ and PMaxQ have different semantics, their PWS-quality function has a common form [i.e., Eq. (3)]. Evaluating the x-form of PMaxQ needs some preprocessing, by sorting the tuples within the same x-tuple according to the querying attributes. An example of tuples sorted in this way is shown in Table 1.

For both queries, the x-form can be computed by iterating on the whole table of x-tuples, in  $O(m)$  times. This is because Eq. (3) is just the sum of  $m$  items, given that the values of  $g(k, D, Q)$  are available. For PMaxQ, an additional average cost of  $O(m \log \frac{n}{m})$  may be needed to sort the tuples. This is still faster than using an exponential number of PW-result probability values to evaluate the PWS-quality. The x-form is also useful to solve the data cleaning problem, to be presented in Sect. 5. We next show a useful fact.

**Lemma 2.**  $g(k, D, Q) < 0$  if and only if there exists  $t_i \in \tau_k$  such that  $p_i \in (0, 1)$ . Otherwise,  $g(k, D, Q) = 0$ .

*Proof.* First of all,  $g(k, D, Q)$  is obviously zero if  $p_i = 0$ , for  $\forall t_i \in \tau_k$ , which means no tuples of this x-tuple can satisfy the query  $Q$ . Secondly, the case  $p_i = 1$  implies that  $e_i = 1$ , which means the x-tuple  $\tau_k$  contains only one tuple and must satisfy the query. Therefore,  $\tau_k$  is clean and brings no uncertainty to the query answer. For all other cases, i.e., there exists  $t_i \in \tau_k$  such that  $p_i \in (0, 1)$ , it is not certain whether the x-tuple satisfies the query or not, and  $g(k, D, Q)$  will be nonzero.

Given an x-tuple  $\tau_k$ , the above states that  $g(k, d, Q)$  is less than zero if there exists a tuple  $t_i \in \tau_k$ , such that its qualification probability,  $p_i$ , is neither zero nor one. More importantly, an x-tuple whose tuples' qualification probabilities are either zero or one *does not* need to be included in computing the PWS-quality [Eq. (3)]. Thus, we do not need to examine the whole database. Instead, we can just pick the x-tuples that satisfy the conditions stated in Lemma 2. This is exactly the set of x-tuples whose tuples in the final query answer have qualification probabilities not equal to one. If we are given the query answer (which are produced by the query engine), then the set of x-tuples required to compute the PWS-quality can be derived

easily. As shown in the experiments in [12], the computation of the PWS-quality, by using the x-form, addresses a performance improvement of around 98 %.

The x-forms for PRQ and PMaxQ can also be used by other entity-based queries. Particularly, the x-form of PRQ can be used by other non-rank-based queries, whose selection conditions only involve the attributes of a single tuple. The x-form of the PMaxQ can also be used by MIN and nearest-neighbor queries, by using a different sorting criterion on the query attribute. Specifically, for MIN queries, we can sort the tuples within an x-tuple in ascending order and change the comparison signs accordingly. The x-form for the nearest-neighbor query can be derived by ordering the tuples according to the Euclidean distance of their querying attributes from the query point.

## 5 Algorithms for Probabilistic Database Cleaning

Let us now discuss how the PWS-quality can be used to facilitate the cleaning of uncertain data. Section 5.1 presents the formal definition of this problem. In Sects. 5.2 and 5.3, we describe an efficient solution that can be applied to the queries under study. Several heuristics that provide efficient solutions are presented in Sect. 5.4.

### 5.1 Problem Definition

Recall that our goal is to select the most appropriate set of x-tuples to be cleaned, under a stringent budget, in order to achieve the highest expected quality improvement. Formally, we define an operation called  $\text{clean}(\tau_k)$ :

**Definition 4.** Given an x-tuple  $\tau_k$ ,  $\text{clean}(\tau_k)$  replaces  $\tau_k$  with an x-tuple that contains a single tuple:  $\{\text{ID}_i, v_i, 1, k\}$ , such that  $\text{ID}_i$  and  $v_i$  are the corresponding identifier and querying attribute value of some tuple  $t_i$  that belongs to  $\tau_k$ .

Essentially,  $\tau_k$  becomes “certain” after  $\text{clean}(\tau_k)$  is performed. Only one of the tuples in the original x-tuple is retained, with existential probability changed to one. The value of the new tuple depends on the specific cleaning operation. In Table 1, for example, after  $\text{clean}(a)$  is performed,  $a$  contains a single tuple  $\{a_2, 80, 1, a\}$ , derived from the information of  $a_2$ , with a price of \$80 and existential probability of one.

Cleaning an x-tuple may involve a cost. For example, if we use an x-tuple to represent a sensor value in a sensor-monitoring application, then the cost of cleaning this x-tuple (by probing the sensor to get the latest value) can be the amount of battery power required for that sensor’s value to be shipped to the base station.

We use  $c_k$ , a natural number, to capture the cost of performing  $\text{clean}(\tau_k)$ . We also assume that a query  $Q$  is associated with a *budget* of  $C$  units, where  $C$  is a natural number. This value limits the maximum amount of cleaning effort that can be used to improve the quality of  $Q$ . In the sensor-monitoring example,  $C$  can be the total amount of energy allowed for probing the sensors. The value of  $C$  may be defined based on the amount of system resource available or the priority of the query user in the system.

Our goal is to obtain the set of x-tuples that, under a given budget, yields the most significant expected improvement in PWS-quality. This set of x-tuples is then selected to be cleaned. Specifically, let  $X$  be any set of x-tuples chosen from database  $D$ . Without loss of generality, let  $X = \{\tau_1, \dots, \tau_{|X|}\}$ . Also, let  $\mathbf{t}$  be a “tuple vector” of  $|X|$  dimensions, where the  $k$ th dimension of  $\mathbf{t}$  is a tuple that belongs to the  $k$ th x-tuple of  $X$ . For example, if  $X = \{\tau_1, \tau_2\}$ , where  $\tau_1 = \{t_0, t_3\}$  and  $\tau_2 = \{t_2, t_5\}$ , then two possible values of  $\mathbf{t}$  are  $\{t_0, t_5\}$  and  $\{t_3, t_2\}$ .

Now, let  $D'(\mathbf{t})$  be the new database obtained, after  $\text{clean}(\tau_k)$  is performed on each x-tuple  $\tau_k$  in  $X$ , which produces tuples described in  $\mathbf{t}$ . The expected quality of cleaning a set  $X$  of x-tuples is then equal to

$$E(S(D'(\mathbf{t}), Q)) = \sum_{\mathbf{t} \in \tau_1 \times \dots \times \tau_{|X|}} \prod_{t_i \in \mathbf{t}} e_i \cdot S(D'(\mathbf{t}), Q) \quad (7)$$

For every tuple vector in  $\tau_1 \times \dots \times \tau_{|X|}$ , Eq. (7) calculates the probability that the new database  $D'(\mathbf{t})$  is obtained (i.e.,  $\prod_{t_i \in \mathbf{t}} e_i$ ) and the PWS-quality score of query  $Q$  evaluated on  $D'(\mathbf{t})$  (i.e.,  $S(D'(\mathbf{t}), Q)$ ).

**Definition 5.** Given a query  $Q$ , the *quality improvement* of cleaning a set  $X$  of x-tuples is

$$I(X, D, Q) = E(S(D'(\mathbf{t}), Q)) - S(D, Q) \quad (8)$$

Our problem can now be formulated as follows:

**Definition 6 (The Data Cleaning Problem).** Given a budget of  $C$  units and a query  $Q$ , choose a set  $X$  of x-tuples from  $D$  such that  $I(X, D, Q)$  attains the highest value.

A straightforward way of solving this problem is to obtain the powerset of all x-tuples in  $D$ . For each element (a set  $X$  of x-tuples) of the powerset, we test whether the total cost of cleaning the x-tuples in  $X$  exceeds the budget  $C$ . Among those that do not, we select the set of x-tuples whose quality improvement is the highest.

This solution is inefficient for two reasons. First, given a set  $X$  of x-tuples, computing equation (8) requires the consideration of all tuple vectors of  $X$ , which are the combinations of tuples selected from the x-tuples in  $X$ . Second, the number of sets of x-tuples to be examined is exponential. We tackle the first problem in Sect. 5.2. The second problem is addressed in Sect. 5.3.

## 5.2 Evaluating Quality Improvement

Equation (8) can be computed more easily by using the x-form of PWS-quality, as shown by the following lemma.

**Lemma 3.** *The quality improvement of cleaning a set  $X$  of  $x$ -tuples is*

$$I(X, D, Q) = - \sum_{k=1}^{|X|} g(k, D, Q) \quad (9)$$

where  $g(k, D, Q)$  is given by Eqs. (4) and (5), for PRQ and PMaxQ, respectively.

*Proof.* By using the x-form [Eq. (3)], we can rewrite  $E(S(D'(\mathbf{t}), Q))$  as

$$\sum_{k=1}^{|X|} E(g(k, D'(\mathbf{t}), Q)) + \sum_{k=|X|+1}^m E(g(k, D'(\mathbf{t}), Q)) \quad (10)$$

For both PRQ and PMaxQ, we claim that

$$g(k, D'(\mathbf{t}), Q) = 0, \text{for } k = 1, \dots, |X| \quad (11)$$

$$E(g(k, D'(\mathbf{t}), Q)) = g(k, D, Q), \text{for } k = |X| + 1, \dots, m \quad (12)$$

We now briefly show that the above two claims are correct for PRQ.<sup>2</sup> First, notice that the new database  $D'(t)$  contains a single tuple for every  $\tau_k \in X$ , whose existential probability is 1, and qualification probability is either 0 or 1. Using this fact and Eq. (4), we can see Eq. (11) is true for every  $k \in [1, |X|]$ . For Eq. (12), observe that  $g(k, D'(\mathbf{t}), Q)$  is just some function [Eq. (4)] of  $p_i$ 's and  $e_i$ 's for  $t_i \in \tau_k$ , where  $\tau_k \notin X$ . As discussed in Sect. 3.2, the value of  $p_i$  for PRQ is either  $e_i$  or zero. Since these values of  $p_i$ 's and  $e_i$ 's are not changed by any cleaning operations on the x-tuples in  $X$ , Eq. (12) holds for  $k = |X| + 1, \dots, m$ .

By using Eqs. (11) and (12), Eq. (10) can be written as  $\sum_{k=|X|+1}^m g(k, D, Q)$ . Together with Eqs. (8) and (3), we can see that Eq. (9) is correct.

Equation (9) reveals three important facts. First, the quality improvement,  $I(X, D, Q)$ , is nonnegative (since  $g(k, D, Q)$  is nonpositive). This implies that the expected quality monotonically increases with the performance of the  $\text{clean}(\tau_k)$  operation. Second, the task of computing  $I(X, D, Q)$  is made easier (compared with Eq. (8)), since  $g(k, D, Q)$  can be computed in polynomial time. If these  $g$  values have been stored (e.g., in a lookup table) during the process of computing the x-form of the PWS-quality [Eq. (3)], then  $I(X, D, Q)$  can be evaluated by a table

---

<sup>2</sup>The proof of PMaxQ is similar, and it can be found in [12].

lookup. Third, Eq. (9) can be applied to both PRQ and PMaxQ, since  $g(k, D, Q)$  have been derived for both queries in Sect. 4.2. Let us see how these results can be used to develop an efficient data cleaning algorithm.

### 5.3 An Optimal and Efficient Data Cleaning Algorithm

We now address the second question: to find out the set  $B$  of x-tuples that leads to the optimal expected quality improvement in PWS-quality, is it possible to avoid enumerating all the combinations of x-tuples in the whole database? To answer this, we first state the following lemma:

**Lemma 4.** *For any x-tuple  $\tau_k \in B$ ,  $\tau_k$  must satisfy the condition: there exists  $t_i \in \tau_k$  such that  $(t_i, p_i)$  appears in the final answer of  $Q$ , with  $p_i \in (0, 1)$ .*

*Proof.* Consider an x-tuple  $\tau_j$ , whose tuples' qualification probabilities are either zero or one. We can show that  $\tau_j$  does not need to be included in  $B$ . Suppose by contradiction that  $\tau_j \in B$ . According to Lemma 2,  $g(j, d, Q) = 0$ . By using Lemma 3, we can see that including  $\tau_j$  in  $B$  has no effect on the quality improvement, i.e.,  $I(B, D, Q)$ . Thus, it is unnecessary to include  $\tau_j$  in  $B$ .

In fact, by excluding  $\tau_j$ , the remaining x-tuples that we need to consider for cleaning are those that contain at least a tuple  $t_i$  with the following conditions: (1)  $t_i$  appears in the final query answer, and (2)  $p_i = (0, 1)$ .

For example, for the MAX query evaluated in Table 1, the optimal set  $B$  can be derived from the result of the MAX query (Table 2), which contains the tuples from x-tuples  $a, b$ , and  $c$ , but not  $d$ . Correspondingly,  $B$  is the subset of the x-tuples  $\{a, b, c\}$ . Thus, Lemma 4 reduces the search space to the x-tuples whose tuples appear in the query answer. It also means that the input of our data cleaning algorithm can be the tuples contained in the query answer (cf. Fig. 1).

From now on, we focus on the x-tuples that satisfy the conditions of Lemma 4. Let  $Z$  be the number of these x-tuples. We also use  $\tau_k$  (where  $k = 1, \dots, Z$ ) to denote these  $Z$  x-tuples.

**An Optimization Problem.** We now present an efficient algorithm that provides an optimal solution to the data cleaning problem. This algorithm can be applied to entity-based queries, including PRQ and PMaxQ. We assume the values of  $g(k, D, Q)$  have been obtained for all values of  $k = 1, \dots, Z$ . For notational convenience, we also use  $g_k$  to represent  $g(k, D, Q)$  (since  $D$  and  $Q$  are constant parameters). Then, Definition 6 can be reformulated as an optimization problem  $P(C, M)$ , where  $M = \{\tau_1, \dots, \tau_Z\}$  is the set of candidates to be considered and  $C$  is the budget assigned to the query

**Maximize**

$$\sum_{k=1}^Z b_k \cdot g_k \quad (13)$$

**Subject to**

$$\sum_{k=1}^Z b_k \cdot c_k \leq C \quad (14)$$

Here,  $\{b_k | k = 1, \dots, Z, b_k = 0\}$  is a bit vector of length  $Z$ , encoding the IDs of x-tuple(s) chosen from  $M$  to be cleaned. Particularly,  $b_k = 1$  if x-tuple  $\tau_k$  is selected, and  $b_k = 0$  otherwise. Equation (13) is the total quality improvement for cleaning a set of x-tuples (where  $\tau_k$  is chosen if  $b_k = 1$ ), which is the same as Eq. (9). The optimization constraint is described in Eq. (14), which requires that the total cost of cleaning the set of x-tuples cannot be more than  $C$ . Note that since Eq. (13) (or Eq. (9)) is true for PRQ and PMaxQ, the solution to this problem can be applied to both queries.

We now show that the problem  $P(C, M)$  obeys the *optimal substructure* property, which enables dynamic programming [14]. Let  $W$ , a set of x-tuples, be the optimal solution to  $P(C, M)$ . Consider the problem  $P(C - c_a, M - \{\tau_a\})$ , where  $\tau_a$  is some x-tuple in  $M$ . Let  $W' = W - \{\tau_a\}$ . Then,  $W'$  must also be an optimal solution to  $P(C - c_a, M - \{\tau_a\})$ . To see this, suppose  $W''$  (where  $W'' \neq W'$ ) is the optimal solution to  $P(C - c_a, M - \{\tau_a\})$ . Then  $W^\# = W'' \cup \{\tau_a\}$  is also a solution to  $P(C, M)$ . Since  $W''$  is better than  $W'$ , the quality improvement for  $W''$  (obtained by adding  $g_a$  to the quality improvement of  $W''$  using Eq. (13)) must also be higher than  $W$ . This violates the assumption that  $W$  is the optimal solution to  $P(C, M)$ . Therefore,  $W'$  must be the optimal solution to  $P(C - c_a, M - \{\tau_a\})$ .

**A Dynamic-Programming Solution.** We now explain how to use a DP algorithm to solve  $P(C, M)$ . Let  $F(i, b)$  be the maximum improvement in quality, by using budget  $b$  to clean the first  $i$  x-tuples. Our objective is to maximize  $F(Z, C)$ , which can be recursively evaluated:

$$F(i, b) = \max \begin{cases} F(i - 1, b) \\ F(i - 1, b - c_i) + g_i, \text{ if } b \geq c_i \end{cases} \quad (15)$$

In the first case of Eq. (15), we do not select the  $i$ th x-tuple to clean. In the second case, the  $i$ th tuple is chosen to clean, so that the quality is improved by an amount of  $g_i$ . To do this, we use a cost of  $c_i$ ; also, a budget of  $b - c_i$  must be available for the first  $i - 1$  x-tuples. Initially,  $F(0, b) = 0$  for  $b = 0, 1, \dots, C$ .

Algorithm 1 shows the details of computing the largest value of  $F(Z, C)$ . An array  $S[i, j]$ , which is a vector with a length of  $Z$ , is used to record the selected x-tuple set in each state: if  $S[i, j][k] = 1$ , the  $k$ th x-tuple is chosen to be cleaned to achieve the optimal  $F[i, j]$ . After the algorithm is completed,  $S[Z, C]$  stores the

**Algorithm 1:** DP

---

**Input:** Cleaning costs  $(c_1, c_2, \dots, c_Z)$ , quality gain  $(g_1, g_2, \dots, g_Z)$ , budget  $C$   
**Output:** Quality Improvement I, selected set of x-tuples  $S$

```

1 for  $i \leftarrow 0$  to  $C$  do
2   |  $F[0, i] \leftarrow 0$ 
3 end
4 for  $i \leftarrow 1$  to  $Z$  do
5   | for  $j \leftarrow 1$  to  $C$  do
6     |   | if  $j \geq c_i$  and  $F[i - 1, j - c_i] + g_i > F[i - 1, j]$  then
7       |   |   |  $F[i, j] \leftarrow F[i - 1, j - c_i] + g_i$ 
8       |   |   |  $S[i, j] \leftarrow S[i - 1, j - c_i]$ 
9       |   |   |  $S[i, j][i] \leftarrow 1$ 
10      |   | end
11      |   | else
12        |   |   |  $F[i, j] \leftarrow F[i - 1, j]$ 
13        |   |   |  $S[i, j] \leftarrow S[i - 1, j]$ 
14      |   | end
15    | end
16 end
17 return  $F[Z, C], S[Z, C]$ 
```

---

selected x-tuples for the cleaning task under the given budget  $C$ . The time and space complexities of Algorithm 1 are both equal to  $O(Z^2 \times C)$ .

## 5.4 Heuristics for Data Cleaning

To further improve the efficiency of data cleaning, we have developed three other heuristics:

1. *Random*: This is the simplest heuristic, where x-tuples are selected randomly until the query budget is exhausted.
2. *MaxQP*: Compute the qualification probability  $P_k$  for each x-tuple  $\tau_k$ , using Eq. 2. Then, choose the x-tuples in descending order of  $P_k$  (where  $P_k \neq 1$ ) until the total cost exceeds  $C$ . The rationale is that selecting x-tuples with higher qualification probabilities may have a better effect on the PWS-quality than those with small values.
3. *Greedy*: Define  $f_k = \frac{g_k}{c_k}$ . Select the x-tuples with the highest values of  $f_k$  such that the maximum total cost is less than  $C$ . Intuitively,  $f_k$  is the quality improvement of  $clean(\tau_k)$  per unit cost. The choice of x-tuples is thus decided by the amount of quality improved and the cost required.

The complexity of the random algorithm is  $O(1)$ . For the MaxQP and the Greedy, their running times are dominated by the effort of sorting  $P_k$  and  $f_k$ , respectively. By using quick sort, their time complexity is  $O(Z \log Z)$ . Notice that the MaxQP and the Greedy can be extended to support large query answer sets. In particular, if the number of x-tuples to be considered by the data cleaning algorithm is too large

to be stored in the main memory, disk-based algorithms (e.g., [27]) can be used to sort the  $x$ -tuples. Then, the  $x$ -tuples that rank the highest can be retrieved.

We have conducted detailed experiments to test the effectiveness and performance of these strategies. We found that while DP achieves the highest cleaning quality, it needs a lot of time to run. Among the heuristics proposed, the Greedy algorithm attains a cleaning effectiveness closest to that of the DP. Moreover, its running time is low. Thus, the Greedy algorithm is both effective and efficient. Interesting readers are referred to [12] for the details of these experiments.

## 6 Conclusions and Future Work

The management of uncertain and probabilistic databases has become an important topic in emerging applications. In this chapter, we investigated a cleaning problem for these databases, with the goal of optimizing the expected quality improvement under a limited budget. To accomplish this task, we designed the PWS-quality metric to quantify query answer ambiguities. We showed how PWS-quality can be efficiently computed for common entity-based queries (PRQ and PMaxQ). We also illustrated that it is possible to develop optimal and efficient solutions that make use of this metric.

We plan to extend our solutions to support other kinds of queries, e.g., top- $k$  and skyline queries. We will also examine other cleaning models. For example, if a cleaning request may not be immediately accomplished, or if a cleaning action results in two or more alternatives, then the solutions discussed here may need to be changed. Another direction is to study how the solution studied here, which supports a single query, can be extended to handle simultaneous query requests. It is also interesting to address how cleaning can be done on databases where the uncertainty of attributes is given by a continuous distribution (e.g., [30, 36]).

## References

1. Agrawal P, Benjelloun O, Sarma AD, Hayworth C, Nabar S, Sugihara T, Widom J (2006) Trio: a system for data, uncertainty, and lineage. In: Proceedings of the VLDB 2006
2. Andritsos P, Fuxman A, Miller R (2006) Clean answers over dirty databases: a probabilistic approach. In: Proceedings of the ICDE 2006
3. Arumugam S, Xu F, Jampani R, Jermaine C, Perez LL, Haas PJ (2010) MCDB-R: risk analysis in the database. In: Proceedings of the VLDB 2010
4. Barbara D, Garcia-Molina H, Porter D (1992) The management of probabilistic data. IEEE TKDE 4(5):487–502
5. Benjelloun O, Sarma A, Halevy A, Widom J (2006) ULDBs: databases with uncertainty and lineage. In: Proceedings of the VLDB 2006
6. Beskales G, Soliman MA, Ilyas IF, Ben-David S (2009) Modeling and querying possible repairs in duplicate detection. In: Proceedings of the VLDB 2009

7. Böhm C, Pryakhin A, Schubert M (2006) The gauss-tree: efficient object identification in databases of probabilistic feature vectors. In: Proceedings of the ICDE 2006
8. Chen J, Cheng R (2008) Quality-aware probing of uncertain data with resource constraints. In: Proceedings of the SSDBM 2008
9. Cheng R, Kalashnikov D, Prabhakar S (2003) Evaluating probabilistic queries over imprecise data. In: Proceedings of the ACM SIGMOD 2003
10. Cheng R, Xia Y, Prabhakar S, Shah R, Vitter JS (2004) Efficient indexing methods for probabilistic threshold queries over uncertain data. In: Proceedings of the VLDB 2004
11. Cheng R, Chen J, Mokbel M, Chow C (2008) Probabilistic verifiers: Evaluating constrained nearest-neighbor queries over uncertain data. In: Proceedings of the ICDE 2008
12. Cheng R, Chen J, Xie X (2008) Cleaning uncertain data with quality guarantees. In: Proceedings of the VLDB 2008
13. Cheng R, Lo E, Yang XS, Luk MH, Li X, Xie X (2010) Explore or exploit?: effective strategies for disambiguating large databases. In: Proceedings of the VLDB 2010
14. Cormen T, Leiserson C, Rivest R, Stein C (2001) Introduction to algorithms. MIT, Cambridge
15. Dalvi N, Suciu D (2004) Efficient query evaluation on probabilistic databases. In: Proceedings of the VLDB 2004
16. de Rougemont M (1995) The reliability of queries. In: Proceedings of the PODS 1995
17. Deshpande A, Guestrin C, Madden S, Hellerstein J, Hong W (2004) Model-driven data acquisition in sensor networks. In: Proceedings of the VLDB 2004
18. Elmagarmid AK, Ipeirotis PG, Verykios VS (2007) Duplicate record detection: a survey. IEEE Trans Knowl Data Eng 19:1–16
19. Gradel E, Gurevich Y, Hirsch C (1998) The complexity of query reliability. In: Proceedings of the PODS 1998
20. Hua M, Pei J, Zhang W, Lin X (2008) Ranking queries on uncertain data: a probabilistic threshold approach. In: Proceedings of the ACM SIGMOD international conference on management of data 2008, pp 673–686
21. Jampani R, Xu F, Wu M, Perez LL, Jermaine C, Haas PJ (2008) MCDB: a Monte Carlo approach to managing uncertain data. In: Proceedings of the 2008 ACM SIGMOD international conference on management of data (SIGMOD’08), pp 687–700
22. Khoussainova N, Balazinska M, Suciu D (2006) Towards correcting input data errors probabilistically using integrity constraints. In: Proceedings of the MobiDE 2006
23. Kriegel H, Kunath P, Renz M (2007) Probabilistic nearest-neighbor query on uncertain objects. In: Proceedings of the DASFAA 2007
24. Li J, Saha B, Deshpande A (2009) A unified approach to ranking in probabilistic databases. In: Proceedings of the VLDB 2009
25. Lian X, Chen L (2008) Monochromatic and bichromatic reverse skyline search over uncertain databases. In: Proceedings of the SIGMOD 2008
26. Liu Z, Sia K, Cho J (2005) Cost-efficient processing of min/max queries over distributed sensors with uncertainty. In: Proceedings of the annual ACM symposium on applied computing (SAC) 2005
27. Nodine M, Vitter J (1995) Greed sort: an optimal sorting algorithm for multiple disks. J ACM 42(4):919–933
28. Olston C, Jiang J, Widom J (2003) Adaptive filters for continuous queries over distributed data streams. In: Proceedings of the SIGMOD 2003
29. Pei J, Jiang B, Lin X, Yuan Y (2007) Probabilistic skylines on uncertain data. In: Proceedings of the VLDB 2007
30. Pfoser D, Jensen C (1999) Capturing the uncertainty of moving-objects representations. In: Proceedings of the SSDBM 1999
31. Re C, Dalvi N, Suciu D (2007) Efficient top-k query evaluation on probabilistic data. In: Proceedings of the ICDE 2007
32. Shannon C (1949) The mathematical theory of communication. University of Illinois Press, Urbana

33. Silberstein A, Braynard R, Ellis C, Munagala K, Yang J (2006) A sampling-based approach to optimizing top-k queries in sensor networks. In: Proceedings of the ICDE 2006
34. Singh S, Mayfield C, Prabhakar S, Shah R, Hambrusch S (2007) Indexing uncertain categorical data. In: Proceedings of the ICDE 2007
35. Singh S, Mayfield C, Shah R, Prabhakar S, Hambrusch SE, Neville J, Cheng R (2008) Database Support for probabilistic attributes and tuples. In: Proceedings of the ICDE 2008
36. Sistla PA, Wolfson O, Chamberlain S, Dao S (1998) Querying the uncertain position of moving objects. In: Temporal databases: research and practice. Springer, Berlin
37. Soliman M, Ilyas I, Chang K (2007) Top-k query processing in uncertain databases. In: Proceedings of the ICDE 2007
38. Soliman MA, Ilyas IF (2009) Ranking with uncertain scores. In: Proceedings of the ICDE 2009
39. Tao Y, Cheng R, Xiao X, Ngai WK, Kao B, Prabhakar S (2005) Indexing multi-dimensional uncertain data with arbitrary probability density functions. In: Proceedings of the VLDB 2005
40. Yi K, Li F, Srivastava D, Kollios G (2008) Efficient processing of top-k queries in uncertain databases. In: Proceedings of the ICDE 2008

# Data Fusion: Resolving Conflicts from Multiple Sources

Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava

**Abstract** Many data management applications, such as setting up Web portals, managing enterprise data, managing community data, and sharing scientific data, require integrating data from multiple sources. Each of these sources provides a set of values, and different sources can often provide conflicting values. To present quality data to users, it is critical to resolve conflicts and discover values that reflect the real world; this task is called *data fusion*. Typically, we expect a true value to be provided by more sources than any particular false one, so we can take the value provided by the largest number of sources as the truth. Unfortunately, a false value can be spread through copying and that makes truth discovery extremely tricky. In this chapter, we consider how to find true values from conflicting information when there are a large number of sources, among which some may copy from others.

We describe a novel approach that considers *copying* between data sources in truth discovery. Intuitively, if two data sources provide a large number of common values and many of these values are unlikely to be provided by other sources (e.g., particular false values), it is very likely that one copies from the other. We apply Bayesian analysis to decide copying between sources and design an algorithm that iteratively detects dependence and discovers truth from conflicting information. We also consider *accuracy* of data sources and *similarity* between values in fusion to further improve the results. We present a case study on real-world data showing that

---

X.L. Dong (✉)

Google Inc., 1600 Amphitheater Pkwy, Mountain View, CA 94043, USA

e-mail: [lunadong@research.att.com](mailto:lunadong@research.att.com)

L. Berti-Equille

IRD - Institut de Recherche pour le Développement, UMR 228 ESPACE-DEV, Maison de la  
Télédétection, 500 rue Jean-François Breton, 34093 MONTPELLIER Cedex 05, FRANCE

e-mail: [Laure.Berti@ird.fr](mailto:Laure.Berti@ird.fr)

D. Srivastava

AT&T Labs-Research, 180 Park Ave., Florham Park, NJ 07932, USA

e-mail: [divesh@research.att.com](mailto:divesh@research.att.com)

the described algorithm can significantly improve accuracy of truth discovery and is scalable when there are a large number of data sources.

## 1 Introduction

The amount of useful information available on the Web has been growing at a dramatic pace in recent years. In a variety of domains, such as science, business, technology, arts, entertainment, politics, government, sports, and tourism, there are a huge number of data sources that seek to provide information to a wide spectrum of information users. In addition to enabling the availability of useful information, the Web has also eased the ability to publish and spread false information across multiple sources. For example, an obituary of Apple founder Steve Jobs was published and sent to thousands of corporate clients on August 28, 2008, before it was retracted.<sup>1</sup> Such false information can often result in considerable damage; for example, the recent incorrect news about United Airlines filing for a second bankruptcy sent its shares tumbling, before the error was corrected.<sup>2</sup> The Web also makes it easy to rapidly spread rumors, which take a long time to die down. For example, the rumor from the late 1990s that the MMR vaccine given to children in Britain was harmful and linked to autism caused a significant drop in MMR coverage, leading autism experts to spend years trying to dispel the rumor.<sup>3</sup> Similarly, the upcoming experiments at the Large Hadron Collider (LHC) have sparked fears among the public that the LHC particle collisions might produce dangerous microscopic black holes that may mean the end of the world.<sup>4</sup>

Widespread availability of conflicting information (some true, some false) makes it hard to separate the wheat from the chaff. Simply using the information that is asserted by the largest number of data sources is clearly inadequate since biased (and even malicious) sources abound, and plagiarism (i.e., copying without proper attribution) between sources may be widespread. How can one find good answers to queries in such a “bad world?” Due to the evident need for practical solutions, topics such as lineage tracking [6–8] and source attribution [1, 3, 5, 9, 17, 20, 21, 27] have been widely studied. *Data fusion* is a promising approach in this space that aims at resolving conflicts from different sources and finds values that reflect the real world.

In this chapter, we describe how we find true values from conflicting information when there are a large number of sources, among which some may copy from others. First, our techniques consider trustworthiness of the sources and give more trust to sources that are more accurate. Second, we determine copying between data sources

<sup>1</sup> [http://www.telegraph.co.uk/news/newstopics/howaboutthat/2638481/Steve-Jobs-obituary\published-by-Bloomberg.html](http://www.telegraph.co.uk/news/newstopics/howaboutthat/2638481/Steve-Jobs-obituary\.published-by-Bloomberg.html).

<sup>2</sup> <http://gawker.com/5047763/how-robots-destroyed-united-airlines>.

<sup>3</sup> <http://www.guardian.co.uk/society/2008/apr/12/health.children>.

<sup>4</sup> [http://en.wikipedia.org/wiki/Large\\_Hadron\\_Collider#Safety\\_of\\_particle\\_collisions](http://en.wikipedia.org/wiki/Large_Hadron_Collider#Safety_of_particle_collisions).

**Table 1** The motivating example: five data sources provide information on the affiliations of five researchers

	$S_1$	$S_2$	$S_3$	$S_4$	$S_5$
<i>Stonebraker</i>	MIT	Berkeley	MIT	MIT	MS
<i>Dewitt</i>	MSR	MSR	UWisc	UWisc	UWisc
<i>Bernstein</i>	MSR	MSR	MSR	MSR	MSR
<i>Carey</i>	UCI	AT&T	BEA	BEA	BEA
<i>Halevy</i>	Google	Google	UW	UW	UW

Only  $S_1$  provides all true values

and downweight copied values in truth discovery. We next illustrate our solution using an example.

*Example 1.* Consider the five data sources in Table 1. They provide information on affiliations of five researchers, and only  $S_1$  provides all correct data. Sources  $S_4$  and  $S_5$  copy their data from  $S_3$ , and  $S_5$  introduces certain errors during copying.

First consider the three sources  $S_1$ ,  $S_2$ , and  $S_3$ . For all researchers except *Carey*, a majority voting on data provided by these three sources can find the correct affiliations. For *Carey*, these sources provide three different affiliations, resulting in a tie. However, if we take into account that the data provided by  $S_1$  is more accurate (among the rest of the four researchers,  $S_1$  provides all correct affiliations, whereas  $S_2$  provides 3 and  $S_3$  provides only 2 correct affiliations), we will consider *UCI* as most likely to be the correct value.

Now consider in addition sources  $S_4$  and  $S_5$ . Since the affiliations provided by  $S_3$  are copied by  $S_4$  and  $S_5$ , naive voting would consider them as the majority and so make wrong decisions for three researchers. Only if we ignore the values provided by  $S_4$  and  $S_5$ , we will be able to again decide the correct affiliations. Note however that identifying the copying relationships is not easy: while  $S_3$  shares five values with  $S_4$  and four values with  $S_5$ ,  $S_1$  and  $S_2$  also share three values, more than half of all values.

## 2 Challenges and Overview of the Solution

Ideally, when applying voting, we would like to give a higher vote to more trustworthy sources and ignore copied information; however, this raises many challenges.

First, we often do not know a priori the trustworthiness of a source, and that depends on how much of its provided data are correct, but the correctness of data, on the other hand, needs to be decided by considering the number and trustworthiness of the providers; thus, it is a chicken-and-egg problem. Indeed, as we show soon, copy detection and truth discovery can also be a chicken-and-egg problem.

Second, in many applications we do not know how each source obtains its data, so we have to discover copiers from a snapshot of data. The discovery is nontrivial: sharing common data does not in itself imply copying—accurate sources can also

share a lot of independently provided correct data. Not sharing a lot of common data does not in itself imply no copying—a copier may copy only a small fraction of data from the original source; even when we decide that two sources are dependent, it is not always obvious which one is a copier.

Third, a copier can also provide some data by itself or verify the correctness of some of the copied data, so it is inappropriate to ignore all data it provides.

In this chapter, we present novel approaches for data fusion. First, we consider *copying* between data sources in truth discovery. Our technique considers not only whether two sources share the same values but also whether the shared values are true or false. Intuitively, for a particular object, there are often multiple distinct false values but usually only one true value. Sharing the same true value does not necessarily imply copying between sources; however, sharing the same false value is typically a low-probability event when the sources are fully independent. Thus, if two data sources share a lot of false values, copying is more likely. In the motivating example (Table 1), if we knew which values are true and which are false, we would suspect copying between  $S_3$ ,  $S_4$ , and  $S_5$ , because they provide the same false values. On the other hand, we would suspect the copying between  $S_1$  and  $S_2$  much less, as they share only true values. Based on this analysis, we describe Bayesian models that compute the probability of copying between pairs of data sources and take the result into consideration in truth discovery.

We also consider *accuracy* in voting: we trust an accurate data source more and give values that it provides a higher weight. This method requires identifying not only if two sources are dependent but also which source is the copier. Indeed, accuracy in itself is a clue of direction of copying: given two data sources, if the accuracy of their common data is highly different from that of one of the sources, that source is more likely to be a copier.

Note that detection of copying between data sources is based on knowledge of true values and accuracy of sources, whereas correctly deciding true values requires knowledge of source copying and accuracy, and deciding accuracy of sources relies on the knowledge of which values are true and which are false. There is an interdependence between them, and we solve the problem by iteratively deciding source copying, discovering truth from conflicting information, and computing accuracy of sources, until the results converge.

In the rest of the chapter, we present how we can leverage source accuracy in data fusion in Sect. 3, present how we can leverage copying relationships in data fusion in Sect. 4, and present a case study of these techniques on a real-world data set in Sect. 5. The techniques we present in this chapter are mainly based on [14], and we shall briefly summarize other techniques in this area.

### 3 Fusing Sources Considering Accuracy

We first formally describe the data fusion problem and describe how we leverage the trustworthiness of sources in truth discovery. In this section we assume no copying between data sources and defer discussion on copying to the next section.

### 3.1 Data Fusion

We consider a set of *data sources*  $\mathcal{S}$  and a set of *objects*  $\mathcal{O}$ . An object represents a particular aspect of a real-world entity, such as the affiliation of a researcher; in a relational database, an object corresponds to a cell in a table. For each object  $O \in \mathcal{O}$ , a source  $S \in \mathcal{S}$  can (but not necessarily) provide a *value*. Among different values provided for an object, one correctly describes the real world and is *true*, and the rest are *false*. In this paper, we solve the following problem: given a snapshot of data sources in  $\mathcal{S}$ , decide the true value for each object  $O \in \mathcal{O}$ .

We note that a value provided by a data source can either be atomic or a set or list of atomic values (e.g., author list of a book). In the latter case, we consider the value as true if the atomic values are correct and the set or list is complete (and order preserved for a list). This setting already fits many real-world applications, and we refer our readers to [32] for solutions that treat a set or list of values as multiple values.

We start our discussion from a core case that satisfies the following two conditions, which we relax later:

- *Uniform false-value distribution*: For each object, there are multiple false values in the underlying domain, and an independent source has the same probability of providing each of them.
- *Categorical value*: For each object, values that do not match exactly are considered as completely different.

Note that this problem definition focuses on *static* information that does not evolve over time, such as authors and publishers of books; directors, actors, and actresses of movies; revenue of a company in past years; presidents of a country in the past; and capitals of countries. Data sources typically rarely update such information, and we consider a snapshot of data from different sources. There are also a lot of information that may evolve over time, such as people's contact information including phone numbers and addresses and businesses that can open or close. We refer our readers to [15] for data fusion for evolving values.

### 3.2 Accuracy of a Source

Let  $S \in \mathcal{S}$  be a data source. The *accuracy* of  $S$ , denoted by  $A(S)$ , is the fraction of true values provided by  $S$ ; it can also be considered as the probability that a value provided by  $S$  is the true value.

Ideally we should compute the accuracy of a source as it is defined; however, in real applications we often do not know for sure which values are true, especially among values that are provided by similar number of sources. Thus, we compute the accuracy of a source as the average probability of its values being true (we

describe how we compute such probabilities shortly). Formally, let  $\bar{V}(S)$  be the values provided by  $S$  and denote by  $|\bar{V}(S)|$  the size of  $\bar{V}(S)$ . For each  $v \in \bar{V}(S)$ , we denote by  $P(v)$  the probability that  $v$  is true. We compute  $A(S)$  as follows:

$$A(S) = \frac{\sum_{v \in \bar{V}(S)} P(v)}{|\bar{V}(S)|}. \quad (1)$$

We distinguish *good* sources from *bad* ones: a data source is considered to be good if for each object it is more likely to provide the true value than any *particular* false value; otherwise, it is considered to be bad. Assume for each object in  $\mathcal{O}$  the number of false values in the domain is  $n$ . Then, in the core case, the probability that  $S$  provides a true value is  $A(S)$  and that it provides a particular false value is  $\frac{1-A(S)}{n}$ . So  $S$  is good if  $A(S) > \frac{1-A(S)}{n}$  (i.e.,  $A(S) > \frac{1}{1+n}$ ). We focus on good sources in the rest of this chapter, unless otherwise specified.

### 3.3 Probability of a Value Being True

Now we need a way to compute the probability that a value is true. Intuitively, the computation should consider both how many sources provide the value and accuracy of those sources. We apply a Bayesian analysis for this purpose.

Consider an object  $O \in \mathcal{O}$ . Let  $\mathcal{V}(O)$  be the domain of  $O$ , including one true value and  $n$  false values. Let  $\bar{S}_o$  be the sources that provide information on  $O$ . For each  $v \in \mathcal{V}(O)$ , we denote by  $\bar{S}_o(v) \subseteq \bar{S}_o$  the set of sources that vote for  $v$  ( $\bar{S}_o(v)$  can be empty). We denote by  $\Psi(O)$  the observation of which value each  $S \in \bar{S}_o$  votes for  $O$ .

To compute  $P(v)$  for  $v \in \mathcal{V}(O)$ , we need to first compute the probability of  $\Psi(O)$  conditioned on  $v$  being true. This probability should be that of sources in  $\bar{S}_o(v)$  each providing the true value and other sources each providing a particular false value:

$$\begin{aligned} Pr(\Psi(O) | v \text{ true}) &= \prod_{S \in \bar{S}_o(v)} A(S) \cdot \prod_{S \in \bar{S}_o \setminus \bar{S}_o(v)} \frac{1 - A(S)}{n} \\ &= \prod_{S \in \bar{S}_o(v)} \frac{n A(S)}{1 - A(S)} \cdot \prod_{S \in \bar{S}_o} \frac{1 - A(S)}{n}. \end{aligned} \quad (2)$$

Among the values in  $\mathcal{V}(O)$ , there is one and only one true value. Assume our *a priori* belief of each value being true is the same, denoted by  $\beta$ . We then have

$$Pr(\Psi(O)) = \sum_{v \in \mathcal{V}(O)} \left( \beta \cdot \prod_{S \in \bar{S}_o(v)} \frac{n A(S)}{1 - A(S)} \cdot \prod_{S \in \bar{S}_o} \frac{1 - A(S)}{n} \right). \quad (3)$$

Applying the Bayes rule leads us to

$$P(v) = \Pr(v \text{ true} | \Psi(O)) = \frac{\prod_{S \in \bar{S}_o(v)} \frac{nA(S)}{1-A(S)}}{\sum_{v_0 \in \mathcal{V}(O)} \prod_{S \in \bar{S}_o(v_0)} \frac{nA(S)}{1-A(S)}}. \quad (4)$$

To simplify the computation, we define the *confidence* of  $v$ , denoted by  $C(v)$ , as<sup>5</sup>

$$C(v) = \sum_{S \in \bar{S}_o(v)} \log \frac{nA(S)}{1-A(S)}. \quad (5)$$

If we define the *accuracy score* of a data source  $S$  as

$$A'(S) = \log \frac{nA(S)}{1-A(S)}, \quad (6)$$

we have

$$C(v) = \sum_{S \in \bar{S}_o(v)} A'(S). \quad (7)$$

So we can compute the confidence of a value by summing up the accuracy scores of its providers. Finally, we can compute the probability of each value as follows:

$$P(v) = \frac{2^{C(v)}}{\sum_{v_0 \in \mathcal{V}(O)} 2^{C(v_0)}}. \quad (8)$$

A value with a higher confidence has a higher probability to be true; thus, rather than comparing vote counts, we can just compare confidence of values. The following theorem shows three nice properties of Eq. (7):

**Theorem 1.** *Equation (7) has the following properties:*

1. *If all data sources are good and have the same accuracy, when the size of  $\bar{S}_o(v)$  increases,  $C(v)$  increases.*
2. *Fixing all sources in  $\bar{S}_o(v)$  except  $S$ , when  $A(S)$  increases for  $S$ ,  $C(v)$  increases.*
3. *If there exists  $S \in \bar{S}_o(v)$  such that  $A(S) = 1$  and no  $S' \in \bar{S}_o(v)$  such that  $A(S') = 0$ ,  $C(v) = +\infty$ ; if there exists  $S \in \bar{S}_o(v)$  such that  $A(S) = 0$  and no  $S' \in \bar{S}_o(v)$  such that  $A(S') = 1$ ,  $C(v) = -\infty$ .*

---

<sup>5</sup>Note that the confidence of a value is derived from, but not equivalent to, the probability of the value.

*Proof.* We prove the three properties as follows:

1. When all data sources have the same accuracy, they have the same accuracy score. Let  $A'$  be the accuracy score and  $s$  be the size of  $\bar{S}_o(v)$ . Then  $C(v) = s \cdot A'$ , so  $C(v)$  increases with  $s$ .
2. When  $A(S)$  increases for a source  $S$ ,  $A'(S)$  increases as well and so  $C(v)$  increases.
3. When  $A(S) = 1$  for a source  $S$ ,  $A'(S) = \infty$  and  $C(v) = \infty$ . When  $A(S) = 0$  for a source  $S$ ,  $A'(S) = -\infty$  and  $C(v) = -\infty$ .

Note that the first property is actually a justification for the naive voting strategy when all sources have the same accuracy. The third property shows that we should be careful not to assign very high or very low accuracy to a data source, which has been avoided by defining the accuracy of a source as the average probability of its provided values.

*Example 2.* Consider  $S_1$ ,  $S_2$ , and  $S_3$  in Table 1 and assume their accuracies are 0.97, 0.6, and 0.4, respectively. Assuming there are 5 false values in the domain (i.e.,  $n = 5$ ), we can compute the accuracy score of each source as follows: for  $S_1$ ,  $A'(S_1) = \log \frac{5*0.97}{1-0.97} = 4.7$ ; for  $S_2$ ,  $A'(S_2) = \log \frac{5*0.6}{1-0.6} = 2$ ; and for  $S_3$ ,  $A'(S_3) = \log \frac{5*0.4}{1-0.4} = 1.5$ .

Now consider the three values provided for *Carey*. Value *UCI* thus has confidence 8, *AT&T* has confidence 5, and *BEA* has confidence 4. Among them, *UCI* has the highest confidence and so the highest probability to be true. Indeed, its probability is  $\frac{2^8}{2^8+2^5+2^4+(5-2)*2^0} = 0.9$ .

### 3.4 Iterative Algorithm

Once we know the confidence of each value, we can choose the one with the highest confidence as the true value. However, computing value confidence requires knowing accuracy of data sources, whereas computing source accuracy requires knowing value probability. There is an interdependence between them, and we solve the problem by computing them iteratively.

In particular, we discover true values from conflicting information provided by multiple data sources as follows.

Algorithm ACCU:

1. Initialize the same accuracy (0.8) to each source.
2. For each source, compute its accuracy score by Eq. (6).
3. For each value, add up the accuracy scores of its providers as its confidence [Eq. (7)].
4. For each value, compute its probability by applying Eq. (8).
5. For each source, take the average probability of its provided values as its accuracy [Eq. (1)].
6. If the accuracies of the sources converge, for each object, output the value with the highest confidence; otherwise, go back to Step 2.

Note that ACCU may not converge; we stop the process after we detect oscillation of decided true values. In practice it has been observed that when the number of objects is much higher than the number of sources, our algorithm typically converges soon; the results generated by different rounds during oscillation have similar overall quality.

### 3.5 Extensions and Alternatives

**Similarity of Values:** We consider similarity between values. Let  $v$  and  $v'$  be two values that are similar. Intuitively, the sources that vote for  $v'$  also implicitly vote for  $v$  and should be considered when counting votes for  $v$ . For example, a source that claims  $UW$  as the affiliation may actually mean  $UWisc$  and should be considered as an implicit voter of  $UWisc$ .

We can extend ACCU by incorporating value similarity as follows. Formally, we denote by  $sim(v, v') \in [0, 1]$  the *similarity* between  $v$  and  $v'$ , which can be computed based on edit distance of strings, difference between numerical values, etc. After computing the confidence of each value of object  $O$ , we adjust them according to the similarities between them as follows:

$$C^*(v) = C(v) + \rho \cdot \sum_{v' \neq v} C(v') \cdot sim(v, v'), \quad (9)$$

where  $\rho \in [0, 1]$  is a parameter controlling the influence of similar values. We then use the adjusted confidence in computation in later rounds.

**Nonuniform Distribution of False Values:** In reality, false values of an object may not be uniformly distributed; for example, an out-of-date value or a value similar to the true value can occur more often than others. We extend ACCU for this situation as follows.

We denote by  $Pop(v|v_t)$  the *popularity* of  $v$  among all false values conditioned on  $v_t$  being true. Then, the probability that source  $S$  provides the correct value (i.e.,  $\Psi_o(S) = v_t$ ) remains  $A(S)$ , but the probability that  $S$  provides a particular incorrect value becomes  $(1 - A(S))Pop(\Psi_o(S)|v_t)$ . Thus, we have

$$\begin{aligned} & Pr(\Psi(O)|v \text{ true}) \\ &= \prod_{S \in \bar{S}_o(v)} A(S) \cdot \prod_{S \in \bar{S}_o \setminus \bar{S}_o(v)} (1 - A(S))Pop(\Psi_o(S)|v) \end{aligned} \quad (10)$$

$$= \prod_{S \in \bar{S}_o(v)} \frac{A(S)}{1 - A(S)} \cdot \prod_{S \in \bar{S}_o \setminus \bar{S}_o(v)} Pop(\Psi_o(S)|v) \cdot \prod_{S \in \bar{S}_o} (1 - A(S)). \quad (11)$$

**Other Ways of Measuring Trustworthiness:** There have been many other ways proposed for measuring the trustworthiness of a source. For example, the measures in [24,25] consider both correctness of data and coverage of provided objects from a

source; one measure in [16, 29] measures the trustworthiness as the cosine similarity between the vector of provided values and the vector of correct values; other measures in [16, 29] also take an average of value confidence but consider not only the provided values but also the values that are not provided (i.e., voted against); techniques in [30, 31] measure source trustworthiness as its accuracy as we do but apply different Bayesian analysis; finally, [32] measures source trustworthiness by specificity and sensitivity in case that there are multiple true values.

## 4 Fusing Sources Considering Copying

Next, we describe how we detect copiers and leverage the discovered copying relationships in data fusion.

### 4.1 Copying Between Sources

We say that there exists *copying* between two data sources  $S_1$  and  $S_2$  if they derive the same part of their data directly or transitively from a common source (can be one of  $S_1$  and  $S_2$ ). Accordingly, there are two types of data sources: *independent sources* and *copiers*.

An *independent source* provides all values independently. It may provide some erroneous values because of incorrect knowledge of the real world, misspellings, etc.

A *copier* copies a part (or all) of data from other sources (independent sources or copiers). It can copy from multiple sources by union, intersection, etc., and as we focus on a snapshot of data, cyclic copying on a particular object is impossible. In addition, a copier may revise some of the copied values or add additional values, though, such revised and added values are considered as independent contributions of the copier.

To make our models tractable, we consider only *direct* copying in copy detection and truth discovery. We discuss at the end of this section how we distinguish transitive copying and co-copying from direct copying.

### 4.2 Copy Detection

We start with copy detection considering only correctness of values. To make the computation tractable, we make the following assumptions in copy detection:

- *Assumption 1 (independent values)*. The values that are independently provided by a data source on different objects are independent of each other.
- *Assumption 2 (independent copying)*. The copying between a pair of data sources is independent of the copying between any other pair of data sources.

- *Assumption 3 (no mutual copying).* There is no mutual copying between a pair of sources; that is,  $S_1$  copying from  $S_2$  and  $S_2$  copying from  $S_1$  do not happen at the same time.

We note that the real world is complex: different sources may represent the same value in different ways, error rates on different data items can be different, errors of certain types may happen more often, copiers can have various copying behaviors, etc. Instead of modeling every possible variant, the basic model we present in detail next captures the most significant aspects of data providing and copying, so are tractable and can avoid overfitting. Indeed, our experiments on real-world data show that it already obtains high accuracy. At the end of this section, we discuss briefly how we can extend the basic model by considering other aspects of data, such as coverage and formatting of data; by considering correlation on copying, such as copying all values associated with the same real-world entity; and by considering indirect copying, including transitive copying and co-copying.

We next describe the basic copy-detection model.

Assume  $\mathcal{S}$  consists of two types of data sources: good independent sources and copiers. Consider two sources  $S_1, S_2 \in \mathcal{S}$ . We apply Bayesian analysis to compute the probability of copying between  $S_1$  and  $S_2$  given observation of their data. For this purpose, we need to compute the probability of the observed data, conditioned on independence or copying between the sources.

Our computation requires several parameters:  $n$  ( $n > 1$ ), the number of false values in the underlying domain for each object;  $c$  ( $0 < c \leq 1$ ), the probability that a value provided by a copier is copied; and  $A(S_1), A(S_2)$ , the accuracies of the sources. Note that in practice, we may not know values of these parameters *a priori* and the values may vary from object to object and from source to source. We bootstrap our algorithms by setting the parameters to default values initially and iteratively refining them by computing the estimated values according to the truth discovery and copy detection results (details given shortly).

In our observation, we are interested in three sets of objects:  $\bar{O}_t$ , denoting the set of objects on which  $S_1$  and  $S_2$  provide the same true value;  $\bar{O}_f$ , denoting the set of objects on which they provide the same false value; and  $\bar{O}_d$ , denoting the set of objects on which they provide different values ( $\bar{O}_t \cup \bar{O}_f \cup \bar{O}_d \subseteq \mathcal{O}$ ). Intuitively, two independent sources providing the same false value are a low-probability event; thus, if we fix  $\bar{O}_t \cup \bar{O}_f$  and  $\bar{O}_d$ , the more common false values that  $S_1$  and  $S_2$  provide, the more likely that they are dependent. On the other hand, if we fix  $\bar{O}_t$  and  $\bar{O}_f$ , the fewer objects on which  $S_1$  and  $S_2$  provide different values, the more likely that they are dependent. We denote by  $\Phi$  the observation of  $\bar{O}_t, \bar{O}_f$ , and  $\bar{O}_d$  and by  $k_t, k_f$ , and  $k_d$  their sizes, respectively. We next describe how we compute the conditional probability of  $\Phi$  based on these intuitions.

We first consider the case where  $S_1$  and  $S_2$  are independent, denoted by  $S_1 \perp S_2$ . Since there is a single true value, the probability that  $S_1$  and  $S_2$  provide the same true value for object  $O$  is

$$Pr(O \in \bar{O}_t | S_1 \perp S_2) = A(S_1) \cdot A(S_2). \quad (12)$$

Under the *uniform-false-value-distribution* condition, the probability that source  $S$  provides a particular false value for object  $O$  is  $\frac{1-A(S)}{n}$ . Thus, the probability that  $S_1$  and  $S_2$  provide the same false value for  $O$  is

$$Pr(O \in \bar{O}_f | S_1 \perp S_2) = n \cdot \frac{1 - A(S_1)}{n} \cdot \frac{1 - A(S_2)}{n} = \frac{(1 - A(S_1))(1 - A(S_2))}{n}. \quad (13)$$

Then, the probability that  $S_1$  and  $S_2$  provide different values on an object  $O$ , denoted by  $P_d$  for convenience, is

$$Pr(O \in \bar{O}_d | S_1 \perp S_2) = 1 - A(S_1)A(S_2) - \frac{(1 - A(S_1))(1 - A(S_2))}{n} = P_d. \quad (14)$$

Following the *independent-values* assumption, the conditional probability of observing  $\Phi$  is

$$Pr(\Phi | S_1 \perp S_2) = \frac{A(S_1)^{k_f} A(S_2)^{k_f} (1 - A(S_1))^{k_f} (1 - A(S_2))^{k_f} P_d^{k_d}}{n^{k_f}}. \quad (15)$$

We next consider the case when  $S_2$  copies from  $S_1$ , denoted by  $S_2 \rightarrow S_1$ . There are two cases where  $S_1$  and  $S_2$  provide the same value  $v$  for an object  $O$ . First, with probability  $c$ ,  $S_2$  copies  $v$  from  $S_1$ , and so  $v$  is true with probability  $A(S_1)$  and false with probability  $1 - A(S_1)$ . Second, with probability  $1 - c$ , the two sources provide  $v$  independently, and so its probability of being true or false is the same as in the case where  $S_1$  and  $S_2$  are independent. Thus, we have

$$Pr(O \in \bar{O}_t | S_2 \rightarrow S_1) = A(S_1) \cdot c + A(S_1) \cdot A(S_2) \cdot (1 - c), \quad (16)$$

$$Pr(O \in \bar{O}_f | S_2 \rightarrow S_1) = (1 - A(S_1)) \cdot c + \frac{(1 - A(S_1))(1 - A(S_2))}{n} \cdot (1 - c). \quad (17)$$

Finally, the probability that  $S_1$  and  $S_2$  provide different values on an object is that of  $S_1$  providing a value independently, and the value differs from that provided by  $S_2$ :

$$Pr(O \in \bar{O}_d | S_2 \rightarrow S_1) = P_d \cdot (1 - c). \quad (18)$$

We compute  $Pr(\Phi | S_2 \rightarrow S_1)$  accordingly; similarly we can also compute  $Pr(\Phi | S_1 \rightarrow S_2)$ . Now we can compute the probability of  $S_1 \perp S_2$  by applying the Bayes rule:

$$\begin{aligned} & Pr(S_1 \perp S_2 | \Phi) \\ &= \frac{\alpha Pr(\Phi | S_1 \perp S_2)}{\alpha Pr(\Phi | S_1 \perp S_2) + \frac{1-\alpha}{2} Pr(\Phi | S_1 \rightarrow S_2) + \frac{1-\alpha}{2} Pr(\Phi | S_2 \rightarrow S_1)}. \end{aligned} \quad (19)$$

Here  $\alpha = \Pr(S_1 \perp S_2)$  ( $0 < \alpha < 1$ ) is the *a priori* probability that two data sources are independent. As we have no *a priori* preference for copy direction, we set the *a priori* probability for copying in each direction as  $\frac{1-\alpha}{2}$ .

Equation (19) has several nice properties that conform to the intuitions we discussed early in this section, formalized as follows:

**Theorem 2.** *Let  $\mathcal{S}$  be a set of good independent sources and copiers. Equation (19) has the following three properties on  $\mathcal{S}$ .*

1. *Fixing  $k_t + k_f$  and  $k_d$ , when  $k_f$  increases, the probability of copying (i.e.,  $\Pr(S_1 \rightarrow S_2 | \Phi) + \Pr(S_2 \rightarrow S_1 | \Phi)$ ) increases.*
2. *Fixing  $k_t + k_f + k_d$ , when  $k_t + k_f$  increases and none of  $k_t$  and  $k_f$  decreases, the probability of copying increases.*
3. *Fixing  $k_t$  and  $k_f$ , when  $k_d$  decreases, the probability of copying increases.*

*Proof.* We prove the three properties assuming each source has accuracy  $1 - \varepsilon$  ( $\varepsilon$  can be considered as the error rate) as follows, and we can extend for the case where each source has a different accuracy. We only need to prove that the opposite holds for  $\Pr(S_1 \perp S_2 | \Phi)$ .

1. Let  $k_0 = k_t + k_f + k_d$ . Then,  $k_d = k_0 - k_t - k_f$ . We have

$$\begin{aligned} & \Pr(S_1 \perp S_2 | \Phi) \\ &= 1 - \left( 1 + \left( \frac{1-\alpha}{\alpha} \right) \left( \frac{1-\varepsilon-c+c\varepsilon}{1-\varepsilon+c\varepsilon} \right)^{k_t} \left( \frac{\varepsilon-c\varepsilon}{cn+\varepsilon-c\varepsilon} \right)^{k_f} \left( \frac{1}{1-c} \right)^{k_0} \right)^{-1}. \end{aligned}$$

As  $0 < c < 1$ , we have  $0 < \frac{1-\varepsilon-c+c\varepsilon}{1-c\varepsilon} < 1$  and  $0 < \frac{\varepsilon-c\varepsilon}{cn+\varepsilon-c\varepsilon} < 1$ . When  $k_t$  or  $k_f$  increases,  $(\frac{1-\varepsilon-c+c\varepsilon}{1-c\varepsilon})^{k_t}$  or  $(\frac{\varepsilon-c\varepsilon}{cn+\varepsilon-c\varepsilon})^{k_f}$  decreases. Thus,  $\Pr(S_1 \perp S_2 | \Phi)$  decreases.

2. Let  $k_c = k_t + k_f$ . Then,  $k_t = k_c - k_f$ . We have

$$\begin{aligned} & \Pr(S_1 \perp S_2 | \Phi) \\ &= 1 - \left( 1 + \left( \frac{1-\alpha}{\alpha} \right) \left( \frac{1-\varepsilon}{1-\varepsilon+c\varepsilon} \right)^{k_c} \left( \frac{\varepsilon(1-\varepsilon+c\varepsilon)}{(1-\varepsilon)(cn+\varepsilon-c\varepsilon)} \right)^{k_f} \left( \frac{1}{1-c} \right)^k \right)^{-1}. \end{aligned}$$

Because  $\varepsilon < \frac{n}{n+1}$ ,  $\varepsilon(1-\varepsilon+c\varepsilon) < (1-\varepsilon)(cn+\varepsilon-c\varepsilon)$ . Thus, when  $k_f$  increases,  $(\frac{\varepsilon(1-c\varepsilon)}{(1-\varepsilon)(n-cn+c\varepsilon)})^{k_f}$  decreases and so  $\Pr(S_1 \perp S_2 | \Phi)$  decreases.

3. Because  $k_d$  increases,  $(\frac{1}{1-c})^{k_d}$  increases, and so  $\Pr(S_1 \perp S_2 | \Phi)$  increases.

*Example 3.* Continue with Ex.1 and consider the possible copying relationship between  $S_1$  and  $S_2$ . We observe that they share no false values (all values they share are correct), so copying is unlikely. With  $\alpha = 0.5$ ,  $c = 0.2$ ,  $A(S_1) = 0.97$ , and  $A(S_2) = 0.6$ , the Bayesian analysis goes as follows.

We start with computation of  $Pr(\Phi|S_1 \perp S_2)$ . We have  $Pr(O \in \bar{O}_t|S_1 \perp S_2) = 0.97 * 0.6 = 0.582$ . There is no object in  $\bar{O}_f$ , and we denote by  $P_d$  the probability  $Pr(O \in \bar{O}_f|S_1 \perp S_2)$ . Thus,  $Pr(\Phi|S_1 \perp S_2) = 0.582^3 * P_d^2 = 0.2P_d^2$ .

Next consider  $Pr(\Phi|S_1 \rightarrow S_2)$ . We have  $Pr(O \in \bar{O}_t|S_1 \rightarrow S_2) = 0.8 * 0.6 + 0.2 * 0.582 = 0.6$  and  $Pr(O \in \bar{O}_f|S_1 \rightarrow S_2) = 0.2P_d$ . Thus,  $Pr(\Phi|S_1 \rightarrow S_2) = 0.6^3 * (0.2P_d)^2 = 0.008P_d^2$ . Similarly,  $Pr(\Phi|S_2 \rightarrow S_1) = 0.028P_d^2$ .

According to Eq. (19),  $Pr(S_1 \perp S_2|\Phi) = \frac{0.5*0.2P_d^2}{0.5*0.2P_d^2+0.25*0.008P_d^2+0.25*0.028P_d^2} = 0.92$ , so independence is very likely.

### 4.3 Independent Vote Count of a Value

We have described how we decide if two sources are dependent. However, even if a source copies from another, it is possible that it provides some of the values independently, so it would be inappropriate to treat these values as copied values and ignore them. We next describe how to count the *independent* vote for a particular value. We start with ideal vote count assuming all sources have the same accuracy, then describe an approximation, and finally describe how to combine the independent vote count with source accuracy.

#### 4.3.1 Ideal Vote Count

We start from the case where we know deterministically the copying relationship between sources and discuss probabilistic copying subsequently. Consider a specific value  $v$  for a particular object  $O$  and let  $\bar{S}_o(v)$  be the set of data sources that provide  $v$  on  $O$ . We can draw a *copying graph*  $G$ , where for each  $S \in \bar{S}_o(v)$ , there is a node and for each  $S_1, S_2 \in \bar{S}_o(v)$  where  $S_1$  copies from  $S_2$ , there is an edge from  $S_1$  to  $S_2$ .

For each  $S \in \bar{S}_o(v)$ , we denote by  $d(S, G)$  the out-degree of  $S$  in  $G$ , corresponding to the number of data sources from which  $S$  copies. If  $d(S, G) = 0$ ,  $S$  is independent and its vote count for  $v$  is 1. Otherwise, for each source  $S'$  that  $S$  copies from,  $S$  provides a value independently of  $S'$  with probability  $1 - c$ . According to the *independent-copying* assumption, the probability that  $S$  provides  $v$  independently of any other source is  $(1 - c)^{d(S, G)}$  and the total vote count of  $v$  with respect to  $G$  is

$$V(v, G) = \sum_{S \in \bar{S}_o(v)} (1 - c)^{d(S, G)}. \quad (20)$$

However, recall that Eq. (19) computes only a probability of copying in each direction. Thus, we have to enumerate all possible copying graphs and take the sum of the vote count with respect to each of them, weighted by the probability of the graph. Let  $\bar{D}_o$  be the set of possible copying between sources in  $\bar{S}_o(v)$ , and we denote the probability of  $D \in \bar{D}_o$  by  $p(D)$ . Consider a subset  $\bar{D} \subseteq \bar{D}_o$  of  $m$



**Fig. 1** Copying graphs with a copying between  $S_1$  and  $S_3$  and one between  $S_2$ , and  $S_3$ , where  $S_1$ ,  $S_2$ , and  $S_3$  provide the same value on an object

copyings. According to the *independent-copying* assumption, the probability that all and only copying relationships in  $\bar{D}$  hold is

$$\Pr(\bar{D}) = \prod_{D \in \bar{D}} p(D) \prod_{D \in \bar{D}_o - \bar{D}} (1 - p(D)). \quad (21)$$

As each copying can have one of the two directions, there are up to  $2^m$  acyclic copying graphs with this set of copying relationships. Intuitively, the more independent sources in a graph, the less likely that all sources in the graph provide the same value. By applying Bayesian analysis, we can compute the probability of each graph. We skip the equations for space reasons and illustrate the computation of vote count in the following example:

*Example 4.* Consider three data sources  $S_1$ ,  $S_2$  and  $S_3$  that provide the same value  $v$  on an object. Assume  $c = 0.8$  and between each pair of sources the probability of copying is 0.4 (0.2 in each direction). We can compute  $v$ 's vote count by enumerating all possible copying graphs:

- There is 1 graph with no copying. All sources are independent so the vote count is  $1 + 1 + 1 = 3$ . The probability of this graph is  $(1 - 0.4)^3 = 0.216$ .
- There are 6 graphs with only one copying. The total probability of graphs that contain a particular copying is  $(1 - 0.4)^2 * 0.4 = 0.144$ . Each copying has two directions, so the probability of each such graph is  $0.144/2 = 0.072$ . No matter which direction the copying is in, the vote count is  $1 + 1 + 0.2 = 2.2$ .
- There are 12 graphs with two copyings. Figure 1 shows the 4 that contain a copying between  $S_1$  and  $S_3$  and a copying between  $S_2$  and  $S_3$ . The sum of their probabilities is  $(1 - 0.4) * 0.4^2 = 0.096$ . For each of the first three graphs (Fig. 1a-c, each with a single independent source), the vote count is  $1 + 0.2 + 0.2 = 1.4$ , and by applying the Bayes rule, we compute its probability as  $0.32 * 0.096 = 0.03$ . For the last one (Fig. 1d, with two independent sources), the vote count is  $1 + 1 + 0.2^2 = 2.04$  and its probability is  $0.04 * 0.096 = 0.004$ .
- Finally, there are 6 acyclic graphs with three copyings (details ignored to save space), where each has vote count  $1 + 0.2 + 0.2^2 = 1.24$  and probability  $0.4^3/6 = 0.011$ .

The total vote count of  $v$ , computed as the weighted sum, is 2.08.

### 4.3.2 Estimating Vote Count

As there are an exponential number of copying graphs, computing the vote count by enumerating all of them can be quite expensive. To make the analysis scalable, we shall find a way to estimate the vote count in polynomial time.

We estimate a vote count by considering the data sources one by one. For each source  $S$ , we denote by  $\overline{Pre}(S)$  the set of sources that have already been considered and by  $\overline{Post}(S)$  the set of sources that have not been considered yet. We compute the probability that the value provided by  $S$  is independent of any source in  $\overline{Pre}(S)$  and take it as the vote count of  $S$ . The vote count computed in this way is not precise because if  $S$  depends only on sources in  $\overline{Post}(S)$  but some of those sources depend on sources in  $\overline{Pre}(S)$ , our estimation still (incorrectly) counts  $S$ 's vote. To minimize such error, we wish that the probability that  $S$  depends on a source  $S' \in \overline{Post}(S)$  and  $S'$  depends on a source  $S'' \in \overline{Pre}(S)$  be the lowest. Thus, we use a greedy algorithm and consider data sources in the following order:

1. If the probability of  $S_1 \rightarrow S_2$  is much higher than that of  $S_2 \rightarrow S_1$ , we consider  $S_1$  as a copier of  $S_2$  with probability  $Pr(S_1 \rightarrow S_2|\Phi) + Pr(S_2 \rightarrow S_1|\Phi)$  (recall that we assume there is no mutual copying) and order  $S_2$  before  $S_1$ . Otherwise, we consider both directions as equally possible, and there is no particular order between  $S_1$  and  $S_2$ ; we consider such copying *undirectional*.
2. For each subset of sources between which there is no particular ordering yet, we sort them as follows: in the first round, we select a data source that is associated with the undirectional copying of the highest probability ( $Pr(S_1 \rightarrow S_2|\Phi) + Pr(S_2 \rightarrow S_1|\Phi)$ ); in later rounds, each time we select a data source that has the copying with the maximum probability with one of the previously selected sources.

We now consider how to compute the vote count of  $v$  once we have decided an order of the data sources. Let  $S$  be a data source that votes for  $v$ . The probability that  $S$  provides  $v$  independently of a source  $S_0 \in \overline{Pre}(S)$  is  $1 - c(Pr(S_1 \rightarrow S_0|\Phi) + Pr(S_0 \rightarrow S_1|\Phi))$ , and the probability that  $S$  provides  $v$  independently of any data source in  $\overline{Pre}(S)$ , denoted by  $I(S)$ , is

$$I(S) = \prod_{S_0 \in \overline{Pre}(S)} (1 - c(Pr(S_1 \rightarrow S_0|\Phi) + Pr(S_0 \rightarrow S_1|\Phi))). \quad (22)$$

The total vote count of  $v$  is  $\sum_{S \in \bar{S}_o(v)} I(S)$ .

*Example 5.* Continue with Example 4. As all copyings have the same probability, we can consider the data sources in any order. We choose the order of  $S_1, S_2, S_3$ . The vote count of  $S_1$  is 1, that of  $S_2$  is  $1 - 0.4 * 0.8 = 0.68$ , and that of  $S_3$  is  $0.68^2 = 0.46$ . So the estimated vote count is  $1 + 0.68 + 0.46 = 2.14$ , very close to the real one, 2.08.

We formalize properties of the vote-count estimation as follows, showing scalability of our estimation algorithm:

---

**Algorithm 2:** ACCUCOPY: Discover true values by considering accuracy of and copying between data sources.

---

```

0: Input:  $\mathcal{S}, \mathcal{O}$ .
    Output: The true value for each object in  $\mathcal{O}$ .
1: Set the accuracy of each source as  $1 - \epsilon$ ;
2: while (accuracy of sources changes && no oscillation of decided true values)
3:   Compute probability of copying between each pair of sources;
4:   Sort sources according to the copyings;
5:   Compute confidence of each value for each object;
6:   Compute accuracy of each source;
7: for each ( $O \in \mathcal{O}$ )
        Among all values of  $O$ , select the one with the highest confidence as the true value;

```

---

**Theorem 3.** *Our vote-count estimation has the following two properties:*

1. *Let  $t_0$  be the ideal vote count of a value and  $t$  be the estimated vote count. Then,  $t_0 \leq t \leq 1.5t_0$ .*
2. *Let  $s$  be the number of sources that provide information on an object. We can estimate the vote count of all values of this object in time  $O(s^2 \log s)$ .*

#### 4.3.3 Combining with Source Accuracy

Finally, when we consider the accuracy of sources, we compute the confidence of  $v$  as follows:

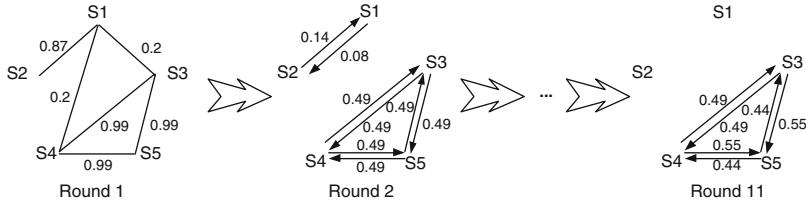
$$C(v) = \sum_{S \in \bar{\mathcal{S}}_v(v)} A'(S)I(S). \quad (23)$$

In the equation,  $I(S)$  is computed by Eq. (22). In other words, we take only the “independent fraction” of the original vote count (decided by source accuracy) from each source.

## 4.4 Iterative Algorithm

We now extend the ACCU algorithm to incorporate analysis of source copying. We need to compute three measures: accuracy of sources, copying between sources, and confidence of values. Accuracy of a source depends on confidence of values, copying between sources depends on accuracy of sources and the true values selected according to the confidence of values, and confidence of values depends on both accuracy of and copying between data sources.

We conduct analysis of both accuracy and copying in each round. Specifically, Algorithm ACCUCOPY starts by setting the same accuracy for each source and the same probability for each value ; then iteratively (1) computes copying based on



**Fig. 2** Probabilities of copyings computed by ACCUCOPY on the motivating example. We only show copyings where the sum of the probabilities in both directions is over 0.1

**Table 2** Accuracy of data sources computed by ACCUCOPY on the motivating example

	$S_1$	$S_2$	$S_3$	$S_4$	$S_5$
Round 1	0.52	0.42	0.53	0.53	0.53
Round 2	0.63	0.46	0.55	0.55	0.41
Round 3	0.71	0.52	0.53	0.53	0.37
Round 4	0.79	0.57	0.48	0.48	0.31
...	...	...	...	...	...
Round 11	0.97	0.61	0.40	0.40	0.21

the confidence of values computed in the previous round, (2) updates confidence of values accordingly, and (3) updates accuracy of sources accordingly, and stops when the accuracy of the sources becomes stable. Note that it is crucial to consider copying between sources from the beginning; otherwise, a data source that has been duplicated many times can dominate the vote results in the first round and make it hard to detect the copying between it and its copiers (as they share only “true” values). Our initial decision on copying is similar to Eq.(19) except considering both the possibility of a value being true and that of the value being false, and we skip details here.

We can prove that if we ignore source accuracy (i.e., assuming all sources have the same accuracy) and there are a finite number of objects in  $\mathcal{O}$ , Algorithm ACCUCOPY cannot change the decision for an object  $O$  back and forth between two different values forever; thus, the algorithm converges.

**Theorem 4.** Let  $\mathcal{S}$  be a set of good independent sources and copiers that provide information on objects in  $\mathcal{O}$ . Let  $l$  be the number of objects in  $\mathcal{O}$  and  $n_0$  be the maximum number of values provided for an object by  $\mathcal{S}$ . The ACCUVOTE algorithm converges in at most  $2ln_0$  rounds on  $\mathcal{S}$  and  $\mathcal{O}$  if it ignores source accuracy.

Once we consider accuracy of sources, ACCUCOPY may not converge: when we select different values as the true values, the direction of the copying between two sources can change and in turn suggest different true values. As in ACCU, we stop the process after we detect oscillation of decided true values. Finally, we note that the complexity of each round is  $O(|\mathcal{O}||\mathcal{S}|^2 \log |\mathcal{S}|)$ .

*Example 6.* Continue with the motivating example. Figure 2 shows the probability of copying, Table 2 shows the computed accuracy of each data source, and Table 3 shows the confidence of affiliations computed for *Carey* and *Halevy*.

**Table 3** Confidence of affiliations computed for *Carey* and *Halevy* in the motivating example

	<i>Carey</i>			<i>Halevy</i>	
	UCI	AT&T	BEA	Google	UW
Round 1	1.61	1.61	2.0	2.1	2.0
Round 2	1.68	1.3	2.12	2.74	2.12
Round 3	2.12	1.47	2.24	3.59	2.24
Round 4	2.51	1.68	2.14	4.01	2.14
...	...	...	...	...	...
Round 11	4.73	2.08	1.47	6.67	1.47

Initially, Line 1 of Algorithm ACCUCOPY sets the accuracy of each source to 0.8. Accordingly, Line 3 computes the probability of copying between sources as shown on the left of Fig. 2. Taking the copying into consideration, Line 5 computes confidence of the values; for example, for *Carey* it computes 1.61 as the confidence of value *UCI* and *AT&T* and 2.0 as the confidence of value *BEA*. Then, Line 6 updates the accuracy of each source to 0.52, 0.42, 0.53, 0.53, and 0.53, respectively, according to the computed value confidence; the updated accuracy is used in the next round.

Starting from the second round,  $S_1$  is considered more accurate and its values are given higher weight. In later rounds, ACCUCOPY gradually increases the accuracy of  $S_1$  and decreases that of  $S_3$ ,  $S_4$ , and  $S_5$ . At the fourth round, ACCUCOPY decides that *UCI* is the correct affiliation for *Carey* and finds the right affiliations for all researchers. Finally, ACCUCOPY terminates at the eleventh round, and the source accuracy it computes converges close to the expected ones (1, 0.6, 0.4, 0.4, 0.2, respectively).

## 4.5 Extensions for Copy Detection

We next describe several extensions for copy detection.

**Considering Other Aspects of Data [13]:** In addition to the values provided by each source, we can also obtain evidence for copying from other aspects of data, such as coverage of the data and formatting of the data. Copying is considered likely if two sources share a lot of objects that are rarely provided by others, if they use common rare formats, and so on.

**Correlated Copying [2, 13]:** The basic model assumes *item-wise independence*, which seldom holds in reality. One can imagine that a copier often copies in one of two modes: (1) it copies data for a subset of entities on a subset of attributes (e.g., title, author list, and publisher of a book), called *per-entity copying*; (2) it copies on a subset of attributes for a set of entities that it provides independently (or entities copied from other sources), called *per-attribute copying*. We can distinguish these two modes in copy detection.

**Global Copy Detection [13]:** The copying discovered by local detection may be due to co-copying or transitive copying. For example, if  $S_3$  copies from  $S_1$  and  $S_2$  and  $S_4$  copies from  $S_3$ , local detection may conclude with  $S_4 \rightarrow S_1$  and  $S_4 \rightarrow S_2$ . The goal of global detection is to fix this problem. The key intuition employed in global detection is that since co-copying and transitive copying can often be inferred from direct copying, we first find a set of copying relationships  $\mathbf{R}$  that significantly influence the rest of the relationships and take them as direct copyings. Then, for each of the remaining copyings, we judge if it is indirect conditioned on  $\mathbf{R}$ ; in other words, in global detection we compute  $Pr(S_1 \rightarrow S_2 | \Phi, \mathbf{R})$  instead of  $Pr(S_1 \rightarrow S_2 | \Phi)$  for pairs outside  $\mathbf{R}$ .

**Dynamic Data [15]:** When we know the update history, we employ a Hidden Markov Model (HMM) to decide whether a source copies from another source and at which moments it copies, exploiting the intuition that the copying relationships can evolve over time, but frequent back-and-forth changes are unlikely.

## 5 A Case Study

We now describe a case study on a real-world data set extracted by searching computer-science books on *AbeBooks.com*. For each book, *AbeBooks.com* returns information provided by a set of online bookstores. Our goal is to find the list of authors for each book. In the data set there are 877 bookstores, 1,263 books, and 24,364 listings (each listing contains a list of authors on a book provided by a bookstore).

We did a normalization of author names and generated a normalized form that preserves the order of the authors and the first name and last name (ignoring the middle name) of each author. On average, each book has 19 listings; the number of different author lists after cleaning varies from 1 to 23 and is 4 on average.

We used a golden standard that contains 100 randomly selected books and the list of authors found on the cover of each book. We compared the fusion results with the golden standard, considering missing or additional authors, misordering, misspelling, and missing first name or last name as errors; however, we do not report missing or misspelled middle names. Table 4 shows the number of errors of different types on the selected books if we apply a naive voting (note that the result author lists on some books may contain multiple types of errors).

We define *precision* of the results as the fraction of objects on which we select the true values (as the number of true values we return and the real number of true values are both the same as the number of objects, the *recall* of the results is the same as the precision). Note that this definition is different from that of accuracy of sources.

### Precision and Efficiency

We compared the following data fusion models on this data set:

- VOTE conducts naive voting.
- SIM conducts naive voting but considers similarity between values.

**Table 4** Different types of errors by naive voting

Missing authors	Additional authors	Misordering	Misspelling	Incomplete names
23	4	3	2	2

**Table 5** Results on the book data set

Model	Precision	Rounds	Time (s)
VOTE	0.71	1	0.2
SIM	0.74	1	0.2
ACCU	0.79	23	1.1
COPY	0.83	3	28.3
ACCUCOPY	0.87	22	185.8
ACCUCOPYSIM	0.89	18	197.5

For each method, we report the precision of the results, the run time, and the number of rounds for convergence. ACCUCOPY and COPY obtain a high precision

- ACCU considers accuracy of sources as we described in Sect. 3 but assumes all sources are independent.
- COPY considers copying between sources as we described in Sect. 4 but assumes all sources have the same accuracy.
- ACCUCOPY applies the ACCUCOPY algorithm described in Sect. 4, considering both source accuracy and copying.
- ACCUCOPYSIM applies the ACCUCOPY algorithm and considers in addition similarity between values.

When applicable, we set  $\alpha = 0.2$ ,  $c = 0.8$ ,  $\varepsilon = 0.2$ , and  $n = 100$ , though, we observed that ranging  $\alpha$  from 0.05 to 0.5, ranging  $c$  from 0.5 to 0.95, and ranging  $\varepsilon$  from 0.05 to 0.3 did not change the results much. We compared similarity of two author lists using 2-g Jaccard distance.

Table 5 lists the precision of results of each algorithm. ACCUCOPYSIM obtained the best results and improved over VOTE by 25.4 %. SIM, ACCU, and COPY each extends VOTE on a different aspect; while all of them increased the precision, COPY increased it the most.

To further understand how considering copying and accuracy of sources can affect our results, we looked at the books on which ACCUCOPY and VOTE generated different results and manually found the correct authors. There are 143 such books, among which ACCUCOPY gave correct authors for 119 books, VOTE gave correct authors for 15 books, and both gave incorrect authors for 9 books.

Finally, COPY was quite efficient and finished in 28.3 seconds. It took ACCUCOPY and ACCUCOPYSIM longer time to converge (3.1 and 3.3 min, respectively), though, truth discovery is often a one-time process, and so taking a few minutes is reasonable.

**Table 6** Bookstores that are likely to be copied by more than ten other bookstores

Bookstore	#Copiers	#Books	Accuracy
Caiman	17.5	1024	0.55
MildredsBooks	14.5	123	0.88
COBU GmbH & Co. KG	13.5	131	0.91
THESAINTBOOKSTORE	13.5	321	0.84
Limelight Bookshop	12	921	0.54
Revaluation Books	12	1091	0.76
Players Quest	11.5	212	0.82
AshleyJohnson	11.5	77	0.79
Powell's Books	11	547	0.55
AlphaCraze.com	10.5	157	0.85
Avg	12.8	460	0.75

For each bookstore, we show the number of books it lists and its accuracy computed by ACCUCOPYSIM

**Table 7** Difference between accuracy of sources computed by our algorithms and the sampled accuracy on the golden standard

	Sampled	ACCUCOPYSIM	ACCUCOPY	ACCU
Average source accuracy	0.542	0.607	0.614	0.623
Average difference	–	0.082	0.087	0.096

The accuracy computed by ACCUCOPYSIM is the closest to the sampled accuracy

### Copying and Source Accuracy:

Out of the 385,000 pairs of bookstores, 2,916 pairs provide information on at least the same 10 books, and among them ACCUCOPYSIM found 508 pairs that are likely to be dependent. Among each such pair  $S_1$  and  $S_2$ , if the probability of  $S_1$  depending on  $S_2$  is over 2/3 of the probability of  $S_1$  and  $S_2$  being dependent, we consider  $S_1$  as a *copier* of  $S_2$ ; otherwise, we consider  $S_1$  and  $S_2$  each has 0.5 probability to be a *copier*. Table 6 shows the bookstores whose information is likely to be copied by more than ten bookstores. On average each of them provides information on 460 books and has accuracy 0.75. Note that among all bookstores, on average each provides information on 28 books, conforming to the intuition that small bookstores are more likely to copy data from large ones. Interestingly, when we applied VOTE on only the information provided by bookstores in Table 6, we obtained a precision of only 0.58, showing that bookstores that are large and copied often actually can make a lot of mistakes.

Finally, we compare the source accuracy computed by our algorithms with that sampled on the 100 books in the golden standard. Specifically, there were 46 bookstores that provide information on more than 10 books in the golden standard. For each of them we computed the *sampled accuracy* as the fraction of the books on which the bookstore provides the same author list as the golden standard. Then, for each bookstore we computed the difference between its accuracy computed by one of our algorithms and the sampled accuracy (Table 7). The source accuracy computed by ACCUCOPYSIM is the closest to the sampled accuracy, indicating

the effectiveness of our model on computing source accuracy and showing that considering copying between sources helps obtain better source accuracy.

## 6 Related Work

Our work is closely related to two research areas: (a) data provenance and (b) trust and authoritativeness of data sources.

**Data Provenance.** Representing and analyzing provenance has been a topic of research since a decade ago [4, 6]. In the literature (e.g., *Open Provenance Model* [23]), provenance is classically modeled as a directed acyclic graph (DAG): the nodes in the DAG represent objects such as files, processes, tuples, and data sets; the edges between two nodes indicate a dependency between the objects. Simmhan et al. [26] provide a taxonomy of provenance characteristics and classify the approaches into data-oriented approaches and process-oriented approaches. Whereas data-oriented approaches focus on data items, process-oriented approaches emphasize information about the processes that produce or consume the data. Bunemann et al. [6–8] identify several open issues for data provenance in the Web era such as (a) obtaining provenance information, (b) citing components of a data resource that may be (components of) another resource in another context, and (c) ensuring integrity of citations under the assumption that cited data resources evolve. Our work can be beneficial to both data-oriented and process-oriented approaches since it collects provenance information, determines copying relationships between dependent data sources, and can be used for DAG generation.

In the context of databases [28] and scientific workflows [11, 12, 33], provenance research usually focuses on the transactions of creation and update of data items by examining data lineage in the query results and data products. In the majority of cases, these approaches consider the sources of a data item that are directly related to the creation process without taking into account possible copying relationships that the data providers may have with each other. With the goal to address this limitation in the context of the Semantic Web, Da Silva et al. [10] propose the *Inference Web* project and describe a provenance infrastructure that supports “the extraction, maintenance and usage of knowledge provenance related to answers of web applications and services.” The term knowledge provenance refers to information about the origin of knowledge and about the reasoning processes used to produce answers. [19] also propose additional dimensions related to the creation and access of data for characterizing provenance information.

**Trust and Authoritativeness of Sources.** Provenance and trust are closely related research topics for many years [9]. Various trust models have been developed emphasizing different characteristics of trust. Artz and Gil [1] provide a comprehensive overview of existing trust models. The most common approach to address trustworthiness in the Web is trust infrastructures that are based on a *Web of*

*Trust* [17]. Approaches such as *PageRank* [5] and *Authorityhub* analysis [21] decide authority based on link analysis [3]. *EigenTrust* [20] and *TrustMe* [27] assign a global trust rating to each data source based on its behavior in a P2P network. While the majority of current approaches consider trustworthiness of data sources, their trustworthiness is not directly related to source accuracy. In addition, they do not consider cases where a data set may have multiple sources, where information providers (re-)publish data aggregated from the original sources, or where inference engines discover implicit facts (or ownership statements) from different sources.

## 7 Summary

In this chapter we present how to improve truth discovery by analyzing accuracy of sources and detecting copying between sources. We describe Bayesian models that discover copiers by analyzing values shared between sources. The results of our models can be considered as a probabilistic database, where each object is associated with a probability distribution of various values in the underlying domain. We described a case study showing that the presented algorithms can significantly improve accuracy of truth discovery and are scalable when there are a large number of data sources.

There are still many open problems for data integration, and here we list a few:

- *Complex fusion functions*: Often, the fusion decision is not based on the conflicting values themselves, but possibly on other data values of the affected tuples, such as a time stamp. In addition, fusion decisions on different attributes of the same tuples often need to coordinate, for instance, in an effort to keep associations between first and last names and not to mix them from different tuples. Providing a language to express such fusion functions and developing algorithms for their efficient execution are open problems.
- *Incremental fusion*: Fusion functions such as voting or average are subject to incorrect results if new conflicting values appear. Techniques, such as retaining data lineage and maintaining simple metadata or statistics, need to be developed to facilitate incremental fusion.
- *Online fusion*: In some applications, it is infeasible to fuse data from different sources in advance either because it is impossible to obtain all data from some sources or because the total amount of data from various sources is huge. In such cases we need to efficiently perform data fusion in an online fashion at the time of query answering. There has been preliminary work in this direction [22], but the work can be extended by considering more types of queries and quality measures.
- *Data lineage*: Database administrators and data owners are notoriously hesitant to merge data and thus lose the original values, in particular if the merged result is not the same as at least one of the original values. Retaining data lineage despite merging is similar to the problem of data lineage through aggregation operators.

Effective and efficient management of data lineage in the context of fusion is yet to be examined.

- *Combining truth discovery and other integration tasks:* The results of data fusion can often benefit other data-integration tasks, such as schema mapping and record linkage. For example, correcting wrong values in some records can help link these records with records that represent the same entity [18]. To obtain the best results in schema mapping, record linkage, and data fusion, we may need to combine them and perform them iteratively.

## References

1. Artz D, Gil Y (2010) A survey of trust in computer science and the semantic web. *J Web Semantics* 5(2)
2. Blanco L, Crescenzi V, Merialdo P, Papotti P (2010) Probabilistic models to reconcile complex data from inaccurate data sources. In: Proceedings of CAiSE
3. Borodin A, Roberts G, Rosenthal J, Tsaparas P (2005) Link analysis ranking: algorithms, theory, and experiments. *ACM TOIT* 5:231–297
4. Bose R, Frew J (2005) Lineage retrieval for scientific data processing: a survey. *ACM Comput Surv* 37(1):1–28
5. Brin S, Page L (1998) The anatomy of a large-scale hypertextual Web search engine. *Comput Netw ISDN Syst* 30(1–7):107–117
6. Buneman P, Cheney J, Tan WC (2008) Curated databases. In: Proceedings of PODS
7. Buneman P, Khanna S, Tan WC (2000) Data provenance: some basic issues. In: Proceedings of the 20th conference on foundations of software technology and theoretical computer science (FST TCS). Springer
8. Buneman P, Khanna S, Tan WC (2001) Why and where: a characterization of data provenance. In: Proceedings of the 8th international conference on database theory (ICDT). Springer
9. Carroll JJ, Bizer C, Hayes P, Stickler P (2005) Named graphs, provenance and trust. In: Proceedings of WWW
10. da Silva PP, McGuinness DL, McCool R (2003) Knowledge provenance infrastructure. *Data Eng Bull* 26(4):26–32
11. Davidson SB, Boulakia SC, Eyal A, Ludaescher B, McPhillips TM, Bowers S, Anand MK, Freire J (2007) Provenance in scientific workflow systems. *IEEE Data Eng Bull* 30(4):44–50
12. Deelman E, Berriman GB, Chervenak A, Corcho O, Groth P, Moreau L (2010) Metadata and provenance management. In: Shoshani A, Rotem D (eds) *Scientific data management: challenges, existing technology, and deployment*. CRC/Taylor and Francis Books (Chapter 12)
13. Dong XL, Berti-Equille L, Hu Y, Srivastava D (2010) Global detection of complex copying relationships between sources. *PVLDB* 3(1):1358–1369
14. Dong XL, Berti-Equille L, Srivastava D (2009) Integrating conflicting data: the role of source dependence. *PVLDB* 2(1):550–561
15. Dong XL, Berti-Equille L, Srivastava D (2009) Truth discovery and copying detection in a dynamic world. *PVLDB* 2(1):562–573
16. Galland A, Abiteboul S, Marian A, Senellart P (2010) Corroborating information from disagreeing views. In: Proceedings of WSDM
17. Golbeck J, Parsia B, Hendler JA (2003) Trust networks on the semantic web. In: Proceedings of the 7th international workshop on cooperative information agents (CIA)
18. Guo S, Dong XD, Srivastava D, Zajac R (2010) Record linkage with uniqueness constraints and erroneous values. *PVLDB* 3(1):417–428

19. Hartig O (2009) Provenance information in the web of data. In: Proceedings of the linked data on the web (LDOW'09), workshop of the world wide web conference (WWW), Madrid
20. Kamvar S, Schlosser M, Garcia-Molina H (2003) The Eigentrust algorithm for reputation management in P2P networks. In: Proceedings of WWW
21. Kleinberg JM (1998) Authoritative sources in a hyperlinked environment. In: SODA
22. Liu X, Dong XL, Ooi BC, Srivastava D (2011) Online data fusion. *PVLDB* 4(11):932–943
23. Moreau L, Clifford B, Freire J, Futrelle J, Gil Y, Groth P, Kwasnikowska N, Miles S, Missier P, Myers J, Plale B, Simmhan Y, Stephan E, Van den Bussche J (2010) The open provenance model core specification (v1.1). Future Generation Computer Systems
24. Pasternack J, Roth D (2010) Knowing what to believe (when you already know something). In: Proceedings of COLING
25. Pasternack J, Roth D (2011) Making better informed trust decisions with generalized fact-finding. In: Proceedings of IJCAI
26. Simmhan Y, Plale B, Gannon D (2005) A survey of data provenance in e-Science. *SIGMOD Rec* 34(3):31–36
27. Singh A, Liu L (2003) TrustMe: anonymous management of trust relationships in decentralized P2P systems (2003). In: IEEE international conference on peer-to-peer computing
28. Tan WC (2007) Provenance in databases: past, current, and future. *IEEE Data Eng Bull* 30(4):3–12
29. Wu M, Marian A (2011) A framework for corroborating answers from multiple web sources. *Inf Syst* 36(2):431–449
30. Yin X, Han J, Yu PS (2007) Truth discovery with multiple conflicting information providers on the Web. In: Proceedings of SIGKDD
31. Yin X, Han J, Yu PS (2008) Truth discovery with multiple conflicting information providers on the web. *IEEE Trans Knowl Data Eng* 20:796–808
32. Zhao B, Rubinstein BIP, Gemmell J, Han J (2012) A Bayesian approach to discovering truth from conflicting sources for data integration. *PVLDB* 5(6):550–561
33. Zhao J (2007) A conceptual model for e-Science provenance. PhD thesis, University of Manchester

## **Part IV**

# **Data Quality in Action**

This part presents case studies from industry covering a variety of industry sectors as well as geographical regions. This part is intended to provide a practical perspective to the management frameworks, technology architectures, and methods and techniques presented in previous parts. The case studies are instrumental in cementing understanding of the importance of the three aspects of the handbook in real data quality initiatives.

The first chapter is on healthcare data and is presented by Heather Richards and Nancy White from the Canadian Institute for Health Information. This chapter highlights the importance of data standards and meta-data management for successful data quality management practices.

The second chapter narrates the story of Shell's global journey towards data quality management maturity. The author, Ken Self, highlights the value of a process-oriented approach for achieving sustainable practices that can have wide ranging impact on a large global organization such as Shell.

The third chapter from Ram Kumar and Robert Logie presents an award winning initiative on data quality improvement within the insurance industry. This chapter outlines the importance of creating an information-centric culture and presents the successes achieved by the approach within SBI General Insurance Limited, India and Insurance Australia Group, Australia.

# Ensuring the Quality of Health Information: The Canadian Experience

Heather Richards and Nancy White

**Abstract** High-quality health information is critical for quality health care and for effective and efficient management of the health care system. This chapter highlights the Canadian experience in the capture and use of health information including a brief introduction to the Canadian health care system and the Canadian Institute for Health Information (CIHI); an overview of CIHI strategies and programs that support quality health information for key stakeholders, including clinicians, health system managers and policymakers; and four case studies from across the health care continuum illustrating data quality strategies in action—prevention, monitoring, feedback and continuous improvement. This chapter concludes with a discussion of two key data quality opportunities for CIHI—the movement toward interoperable electronic health records and data integration. Further information on CIHI, its data quality program and the case studies in this chapter may be found at [www.cihi.ca](http://www.cihi.ca).

## 1 Introduction

High-quality health information is critical for quality health care and for effective and efficient management of the health care system. This chapter highlights the Canadian experience in the capture and use of health information including:

- A brief introduction to the Canadian health care system and the Canadian Institute for Health Information (CIHI)
- An overview of CIHI strategies and programs that support quality health information for key stakeholders, including clinicians, health system managers and policymakers

---

H. Richards (✉) · N. White (retired)  
Canadian Institute for Health Information, Ottawa, ON, Canada  
e-mail: [HRichards@cihi.ca](mailto:HRichards@cihi.ca); [nancyellenwhite@gmail.com](mailto:nancyellenwhite@gmail.com)

- Four case studies from across the health care continuum illustrating data quality strategies in action—prevention, monitoring, feedback and continuous improvement

This chapter concludes with a discussion of two key data quality opportunities for CIHI—the movement toward interoperable electronic health records and data integration. Further information on CIHI, its data quality program and the case studies in this chapter may be found at [www.cihi.ca](http://www.cihi.ca).

## 2 The Canadian Institute for Health Information (CIHI)

In essence, Canada has many health care systems. With the exception of some groups (e.g. First Nations and Inuit, Armed Forces, Royal Canadian Mounted Police) who receive services directly through the federal government, responsibility for the organization and delivery of health services rests with the provinces and territories.

The federal government's health department (i.e. Health Canada) sets and administers national principles for the health care system through the Canada Health Act. The act establishes the criteria and conditions related to publicly insured health services that Canada's thirteen provinces and territories must fulfil in order to receive the full federal cash contribution.

The aim of the Canada Health Act is to ensure that all eligible residents across Canada have reasonable access to "medically necessary" hospital and physician services without charges at the point of service [5]. Coverage for other services, such as long-term care, home care and drug programs, falls under provincial/territorial jurisdiction and varies across the country.

CIHI is an independent, not-for-profit organization established in 1994 by Canada's deputy ministers of health to improve the quality and availability of Canadian health data. Most of CIHI's funding flows from federal, provincial and territorial governments. In collaboration with key stakeholders, CIHI plays a leadership role in setting standards for data collection and for meaningful data analysis to support both front-line and system uses of health information.

CIHI data holdings include information on individuals receiving health care services as well as the organizations and individuals providing the services. Data are captured and submitted to CIHI in a variety of ways depending on the sector and type of information. This diversity requires not only an overarching data quality program but also customized data quality strategies to address the unique challenges of each data holding.

The following sections provide an overview of clinical and health system data at CIHI, highlighting the different data collection processes and data providers that influence the approach to data quality.

## 2.1 Clinical Data

Clinical information about individuals receiving publicly funded services may be captured in real time at the point of care in electronic health records (EHRs), abstracted from a traditional patient chart, or captured by provincial/territorial or other information systems prior to submission to CIHI.

Table 1 provides a brief overview of selected clinical data resources at CIHI and how they are collected and submitted. While all of the clinical data are available at the person level, differences in data standards, data collection and submission processes result in different data quality strengths and challenges.

## 2.2 Health System Data

Health system information is submitted in a variety of ways from a broad range of sources. Health spending data are captured at various levels of aggregation, from individual to hospital, health region and national levels.

Health human resources information is captured by hospitals and health regions, as well as by provincial/territorial professional associations and licensing bodies. Information on medical imaging and wait times are survey-based and require some unique data quality strategies.

Table 2 provides an overview of health system information resources at CIHI. As in the case of the clinical data, the diversity of data providers and processes requires both global and targeted data quality strategies.

## 3 Data Quality at CIHI

*Better data. Better decisions. Healthier Canadians.*

This is the CIHI vision that drives an ongoing quest for optimal data quality—a significant challenge in a rapidly changing environment.

The approach to data quality at CIHI is multifaceted given the many types of data holdings and providers. There is an overarching philosophy and quality mandate spearheaded by CIHI senior management and pervasive throughout the organization.

Data quality is the responsibility of each CIHI staff member, regardless of position or department. Each program area responsible for a data holding plays a key role in continuous improvement of the data. A dedicated data quality department provides support to CIHI staff in the identification and resolution of data quality issues.

**Table 1** Capturing clinical data in Canada

Focus	Type of data	Collection/submission process
Primary care	Person-level clinical/administrative	Captured at the point of care in physician office/clinic. Data are submitted to a voluntary reporting system at CIHI on a periodic basis for each episode of care
	Person-level drug utilization	Captured at point of service in pharmacies for individuals covered by public drug benefit plans. Data are submitted to CIHI through provincial/territorial governments on a periodic basis
Acute care	Person-level clinical/administrative	Diagnoses and interventions abstracted and coded retrospectively from primary clinical record using the Canadian version of the World Health Organization's ICD-10 standard (ICD-10-CA) and CIHI's Canadian Classification of Health Interventions (CCI). Data are submitted by hospitals to CIHI on a periodic basis for all inpatient, emergency and ambulatory discharges
	Person-level registries	Specialized information (e.g. dialysis, trauma, joint replacement) captured at point of care. Data are submitted to CIHI by hospitals and clinics on a periodic basis for all discharges. A longitudinal record is created at CIHI for an individual
Mental health	Person-level clinical/administrative	Captured in acute care abstracts and an organizational survey submitted periodically to CIHI. In Ontario, inpatient data captured at point of care using the RAI-MH <sup>©</sup> . Data are submitted to CIHI on a periodic basis. A longitudinal record is created at CIHI for an individual
Rehabilitation	Person-level clinical/administrative	Captured by clinicians at point of care in hospitals using FIM™ standardized assessment at admission and discharge. Data are submitted to CIHI on a periodic basis for all inpatient episodes of care
Continuing care	Person-level clinical/administrative	Captured by clinicians at point of care in hospitals and residential care facilities (e.g. nursing homes) using standardized assessment (RAI-MDS 2.0 <sup>©</sup> ) administered at intervals throughout the episode of care. Data are submitted to CIHI on a periodic basis. A longitudinal record is created at CIHI for an individual
Home care	Person-level clinical/administrative	Captured by clinicians at point of care using standardized assessment (RAI-HC <sup>©</sup> ) administered at intervals throughout the episode of care. Data are submitted to CIHI by regional health organizations on a periodic basis. A longitudinal record is created at CIHI for an individual

The RAI-MH<sup>©</sup> is a copyright of the Government of Ontario, Ontario Hospital Association and interRAI Corporation. The FIM™ instrument is the property of Uniform Data System for Medical Rehabilitation, a division of UB Foundation Activities, Inc. The RAI-MDS 2.0<sup>©</sup> and RAI-HC<sup>©</sup> are a copyright of interRAI Corporation

**Table 2** Capturing health system data in Canada

Focus	Type of data	Collection/submission process
Health spending	Person-level service and payment data for physicians	Captured at point of service and submitted periodically to CIHI by provincial/territorial governments
	Aggregate financial/administrative data for hospitals and health regions	Captured using a standardized chart of accounts and submitted periodically to CIHI by provincial/territorial governments
	Aggregate health expenditures for Canada	Collected through federal and provincial/territorial governments, insurance companies, workers compensation boards and other sources
Health system resources	Person-level health workforce	Socio-demographic data on physicians collected and submitted to CIHI by provincial/territorial governments. Data on nurses and other selected health professionals (e.g. pharmacists, therapists) are submitted annually from provincial/territorial associations and regulatory authorities
	Aggregate health workforce	Socio-demographic data for 24 health occupations collected through provincial/territorial health professional associations and regulatory authorities who submit periodically to CIHI
	Aggregate wait times	Procedure-specific wait times data captured using standard definitions and submitted periodically to CIHI by provincial/territorial governments
	Aggregate technology survey	Annual volumes and utilization rates of medical imaging equipment in Canadian hospitals and freestanding imaging facilities

However, optimal data quality may only be achieved when all key players along the journey from data capture to data use understand their role and accept accountability for data quality. CIHI collaborates with the many organizations involved along the data supply chain and fosters a sense of stewardship for its quality.

CIHI defines information and data quality in the context of its data users. These include health system planners and managers, health care providers, federal and provincial quality councils, health researchers, professional associations and other health-related organizations. If CIHI data satisfy their specific information needs, then the data are said to be “fit for use”.

This suggests that data need not be “perfect”—an unattainable goal. However, with appropriate knowledge of the strengths and weaknesses of a given data set, a data user can make informed decisions about whether it may be confidently used to answer a particular business or research question.

A number of data quality challenges are ongoing and will be discussed further in subsequent sections of this chapter. In particular, Canada-wide agreement on data and technical standards is difficult given the sheer number and diversity of stakeholders involved in health information.

For some data holdings at CIHI, data may be collected for purposes other than national reporting. For example, health human resource data are collected by professional associations for their own licensing or registration purposes. In this situation, these data are considered “secondary data” at CIHI since the primary purpose for its collection was for licensing purposes rather than CIHI’s purpose for statistical analysis and reporting.

Data quality efforts at CIHI have evolved over the years in response to changes in the health system, technology, policy and privacy landscape.

CIHI’s corporate data quality program includes processes and policies to improve data and information quality, both within CIHI and in the broader health sector. The program is designed to support prevention, early detection and resolution of data issues through a continuous improvement process.

### ***3.1 CIHI Prevention Strategies***

One of the key principles in the CIHI data quality strategy is to avoid the capture and submission of poor quality data. CIHI supports a broad range of prevention strategies.

In order for data to be comparable across Canada, it must be captured and transmitted to CIHI in a standardized way. Tables 1 and 2 provided examples of clinical and health system data standards used in CIHI data holdings.

These standards may be developed by CIHI or adapted from others, including national and international organizations. The following sections provide an overview of CIHI data content standards, the technical standards that facilitate capture and submission, and the services available to those who collect, submit and use the information.

#### **3.1.1 Data Standards**

One of CIHI’s foundational strategies for ensuring quality data is the capture of standardized data elements, whether they are a diagnosis or intervention, a unit of time or cost or any other key piece of information needed by the users.

This requires a rigorous development and testing process to ensure that the standards produce reliable and valid data. To minimize variation due to individual interpretation, the standards must be clear, unambiguous and well understood by those capturing or abstracting the data.

In developing or adapting standards for Canada, CIHI conducts extensive consultation with those who will be collecting and using the data. Tables 1 and 2 highlight

a number of international standards used at CIHI, including those developed by the World Health Organization (WHO) and interRAI. These collaborations ensure the credibility, relevance and usefulness of the information. This in turn provides a strong incentive to submit high-quality data to CIHI and to collaborate on data quality initiatives.

Each program area at CIHI employs subject matter experts from its sector of the health care system. These experts liaise with key stakeholders to maintain and continuously improve the data standards for the evolving health care system.

### **3.1.2 Technical Standards**

Once the data standards are defined and agreed upon by the users, CIHI develops and disseminates technical standards for their electronic capture and submission. These are developed in consultation with key stakeholders, including national and international standard-setting bodies (e.g. ISO, Canada Health Infoway).

For many of CIHI's data holdings, third-party software vendors play an important role in ensuring adherence to both clinical and technical standards.

The CIHI vendor support team regularly issues vendor specifications for each data holding, oversees the vendor testing process and coordinates a feedback and continuous improvement process. CIHI vendor specifications also document data quality audits and edits to be applied prior to the data being accepted by CIHI. These are discussed in more detail in the following section.

For initial submissions to CIHI and on an annual basis thereafter, all vendors are required to successfully submit test files to CIHI that demonstrate conformance with the standards. This testing supports consistency of information across organizations and jurisdictions.

### **3.1.3 System Edits and Audits**

CIHI develops in-house technical solutions to support the capture, submission, analysis and reporting of health data. In the earliest stages of information system design at CIHI, key requirements are identified to support optimal data quality.

Key stakeholders work with CIHI to identify data quality risks and concerns specific to the data holding. CIHI specifies the criteria for rejection of data (e.g. system edits) and for flagging of suspicious data (system audits).

Depending upon the data type and source, some or all of these may be applied at the point of care or data capture, providing optimal prevention by avoiding the capture of erroneous data. The case study in Section 4 on the Continuing Care Reporting System provides additional detail on a point-of-care application.

For other data, which may be submitted as secondary data, the checks are done upon submission to CIHI, triggering the correction and resubmission process.

### **3.1.4 Training and Support**

CIHI provides extensive training and support for clinicians, coders and data managers who have a key role to play along the data supply chain. There are also training programs for health system analysts, managers and policymakers who use the data to support organization- and system-level decision-making.

CIHI training includes face-to-face hands-on workshops, interactive web conferences and self-learning packages. Topics cover data standards, submission and correction processes, data quality and appropriate use of CIHI data and statistical reports.

Additional support is provided through an online query tool that maintains a database of questions regarding the data standards and provides ad hoc responses to client questions as needed. CIHI experts in each program are also available to assist data providers with unique coding or data submission questions.

Many of CIHI's programs offer opportunities for additional client support through web conferences targeted at specific topics of interest, such as difficult situations in coding or interpreting the standards. Others hold a regular web-based open forum to field questions and provide timely advice.

CIHI training and support are available to client jurisdictions through bilateral agreements with provinces and territories so that no individual course fees are charged to participants. With a minimum of administrative process, thousands of health personnel across the country receive the training they need to fulfil their role in promoting data quality.

## **3.2 *CIHI Monitoring and Feedback Strategies***

CIHI employs a wide range of monitoring and feedback strategies, from real-time flagging and reporting of record-level errors aimed at data submitters to periodic, aggregate information targeted at senior managers in participating jurisdictions.

### **3.2.1 Operational Reporting and Corrections**

Timely monitoring of data submissions and a user-friendly process for correction of erroneous data are fundamental to data quality. Most CIHI reporting systems generate web-based feedback reports for data submitters as soon as data are processed. These record-level reports immediately identify records that were not accepted due to data quality edits and those that were accepted with data quality audit flags.

The data submitters for most CIHI data holdings then have a window of opportunity to submit corrected records and improve their data prior to its release for analysis and use.

CIHI also produces web-based data quality reports on a routine basis, some daily and others on a quarterly basis, that provide summary information on errors at organizational levels, informing managers on their performance in a given period and feeding into the continuous improvement process.

### **3.2.2 Data Quality Assessment**

CIHI uses a variety of tools to evaluate the strengths and limitations of its data holdings. CIHI data are subject to routine evaluation using an assessment tool that is part of the CIHI Data Quality Framework [1]. Reabstraction studies, targeted analysis and data mining, where appropriate, also inform the evaluation.

The CIHI Data Quality Framework Assessment Tool articulates five dimensions of data quality: accuracy, timeliness, comparability, usability and relevance. It expands each dimension with a set of characteristics and criteria for assessment as described in Table 3.

CIHI staff use the framework tool to determine the degree to which a given data set meets its criteria for quality. The results of the assessment inform CIHI continuous improvement strategies. They are also an important component of documentation for data users, supporting their decisions regarding fitness for use.

### **3.2.3 Reabstraction Studies**

For CIHI clinical data holdings (e.g. acute care inpatient data) that rely on the coding and abstraction of data from medical charts by health record personnel prior to data submission, the assessment of data quality may include reabstraction studies.

Reabstraction studies determine the degree of measurement error (i.e. consistency, bias) in a given data set. The reabstraction process involves collecting information from source documents (e.g. medical chart) for data that have previously been captured and submitted to CIHI.

The reabstracted data are compared to the original data and causes for any observed differences are investigated. Issues may relate to the quality of the chart documentation, noncompliance with coding standards or directives, cases where different codes were used but either code is acceptable or system errors in cases where data were downloaded incorrectly.

The results provide important feedback for CIHI, data users and providers, and guide targeted strategies for improvement. The second case study in Section 4 provides an example of a reabstraction study conducted by CIHI using emergency department data.

Since reabstraction studies are time consuming and resource intensive, they are used at CIHI to investigate suspected data quality issues and are not part of the annual data quality cycle for abstracted data. Furthermore, their design is not appropriate in cases where clinical data are captured electronically as part of the process of care, where secondary analysis and data mining are used to inform the assessment.

**Table 3** CIHI Data Quality Framework Assessment Tool [1]

Characteristics	Criteria
<b>Accuracy dimension</b>	
Coverage	The population of reference is explicitly stated in all releases Efforts are being made to close the gap between the population of reference and the population of interest Known sources of under- or overcoverage have been documented The frame has been validated by comparison with external and independent sources The rate of under- or overcoverage falls into one of the predefined categories
Capture and collection	CIHI practices that minimize response burden are documented CIHI has documentation of data-provider practices that minimize response burden Practices exist that encourage cooperation for data submission Practices exist that give support to data providers Standard data submission procedures exist and are followed by data providers Data capture quality control measures exist and are implemented by data providers
Unit nonresponse	The magnitude of unit nonresponse is mentioned in the data quality documentation The number of records for responding units is monitored to detect unusual values The magnitude of unit nonresponse falls into one of the predetermined categories
Item (partial) nonresponse	Item nonresponse is identified The magnitude of item nonresponse falls into one of the predetermined categories
Measurement error	The level of measurement error falls into one of the predetermined categories The level of bias is not significant The degree of problems with consistency falls into one of the predetermined categories
Edit and imputation	Validity checks are done for each data element and any invalid data is flagged Edit rules and imputation are logical and applied consistently Edit reports for users are easy to use and understand The imputation process is automated and consistent with the edit rules
Processing and estimation	Documentation for all data processing activities is maintained Technical specifications for the data holding are maintained Changes to a data holding's underlying structure or processing or estimation programs have been tested Raw data, according to the CIHI policy for data retention, is saved in a secure location

(continued)

**Table 3** (continued)

Characteristics	Criteria
	<p>Aggregated statistics from a data holding have been compared, where possible, to similar statistics from another CIHI data holding or external source</p> <p>The variance of the estimate, compared to the estimate itself, is at an acceptable level</p>
<b>Timeliness dimension</b>	
Data currency at the time of release	<p>The difference between the actual date of data release and the end of the reference period is reasonably brief</p> <p>The official date of data release was announced before the release</p> <p>The official date of data release was met</p> <p>Data processing activities are regularly reviewed to improve timeliness</p>
Documentation currency	<p>The recommended data quality documentation was available at the time of data or report release</p> <p>Major reports were released on schedule</p>
<b>Comparability dimension</b>	
Data Dictionary standards	<p>All data elements are evaluated to determine their inclusion within the CIHI Data Dictionary</p> <p>Data elements from a data holding that are contained within the CIHI Data Dictionary conform to dictionary standards</p>
Standardization	<p>Data is collected at the finest level of detail that is practical</p> <p>For any derived data element, the original data element remains accessible</p>
Linkage	<p>Geographical data is collected using Statistics Canada's Standard Geographical Classification</p> <p>Data is collected using a consistent time frame, especially between and within jurisdictions</p> <p>Identifiers are used to differentiate facilities or organizations uniquely for historical linkage</p> <p>Identifiers are used to differentiate persons or machines uniquely for historical linkage</p>
Equivalency	<p>Methodology and limitations for crosswalks and/or conversions are documented</p> <p>The magnitude of issues related to crosswalks and conversions falls into one of the predetermined categories</p>
Historical comparability	<p>Documentation on historical changes to the data holding exists and is easily accessible</p> <p>Trend analysis is used to examine changes in core data elements over time</p> <p>The magnitude of issues associated with comparing data over time falls into one of the predetermined categories</p>
<b>Usability dimension</b>	
Accessibility	<p>A final data set is made available per planned release</p> <p>Standard tables and analyses using standard format and content are produced per planned release or upon request</p> <p>Products are defined, catalogued and/or publicized</p>

(continued)

**Table 3** (continued)

Characteristics	Criteria
Documentation	Current data quality documentation for users exists Current metadata documentation exists A caveat accompanies any preliminary release
Interpretability	A mechanism is in place whereby key users can provide feedback to, and receive notice from, the data holding program area Revision guidelines are available and applied per release
<b><i>Relevance dimension</i></b>	
Adaptability	Mechanisms are in place to keep stakeholders informed of developments in the field The data holding is developed so that future system modifications can be made easily
Value	The mandate of the data holding fills a health information gap The level of usage of the data holding is monitored User satisfaction is periodically assessed

### 3.2.4 Analysis and Data Mining

In data holdings where data are not abstracted, other analytical strategies may be employed. Some CIHI data are captured electronically at the point of care using clinical assessment instruments developed and tested by others for reliability and validity (see Table 1).

CIHI conducts secondary analyses on the interRAI data sets, including tests of convergent validity to evaluate the reliability of key clinical scales over time. Longitudinal analyses look for inconsistencies in items that should be stable across episodes of care for an individual, such as paraplegia or a diagnosis of multiple sclerosis.

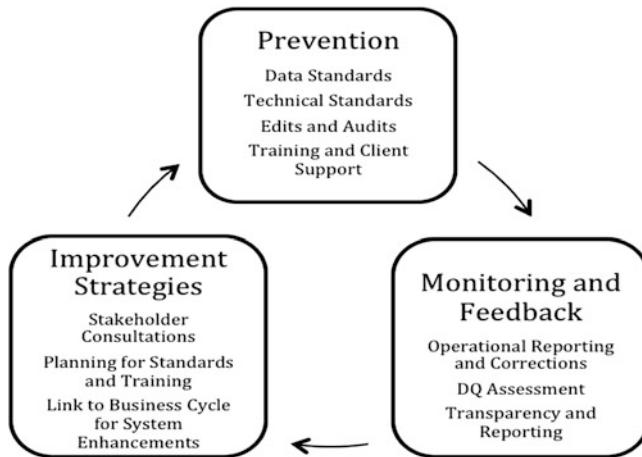
CIHI also conducts targeted data mining activities, often in collaboration with data providers and users, including Ministries of Health. These studies may be conducted to provide support for system funding models or quality report cards where confidence in data quality is critical.

A statistical tool that analyzes CIHI health system financial and activity data to detect anomalies is featured in the third case study in Section 4.

### 3.2.5 Transparency and Reporting

Details of data standards, coverage and data quality (metadata) are documented and available on the website for CIHI data holdings. This documentation supports the appropriate capture, submission, understanding and use of the data.

CIHI also publishes external data quality documentation, based on the Data Quality Framework Assessment Tool, along with other available information that allows users to assess fitness for use.



**Fig. 1** Continuous data quality improvement process

An annual data quality report card designed for provincial/territorial deputy ministers of health provides a snapshot across CIHI data holdings on key indicators such as timeliness, coverage and error rates. Deputies can compare their jurisdiction's progress toward quality data with other provinces and territories and provide high-level direction for improvements.

System managers access web-based CIHI comparative reports on the populations and quality indicators of their organization and benchmark with their peers. Variation in these indicators often triggers data quality reviews and improvements.

And finally, public release of the data sets and use of the data in analysis and public reports provide a strong incentive for jurisdictions to address data quality issues. The more the data are used by jurisdictions in performance reporting, planning and funding, the more attention is paid to its quality. This focus effectively drives ongoing data quality improvements.

### ***3.3 CIHI Continuous Improvement Strategies***

The prevention, monitoring and feedback strategies described above are the foundation for continuous improvement activities at CIHI and in the field as illustrated in Fig. 1.

Data quality improvement is integrated into the annual business cycle for each data holding. CIHI consults regularly with key stakeholders and issues specifications for the data and submission standards, as well as edit and audit rules. Training, client support and reporting programs reflect the current year's standards.

Data quality is monitored and reported throughout the year, with CIHI supporting corrections and resubmissions. The findings of the CIHI data quality monitoring and

assessment activities are then fed into the annual planning process, when changes may be proposed to address identified issues and challenges.

The effects of these changes are monitored in the following year to assess whether further interventions are required, closing the loop on the data quality cycle.

CIHI also supports Advisory Committees for each data holding that include representatives of provinces, territories and health regions, along with selected data providers, subject matter experts and researchers. Meetings generally correspond with the business planning cycle with an annual face-to-face meeting and others by teleconference as required. The committees provide CIHI with important feedback and strategic advice on future enhancements to the standards, data collection processes and reporting tools.

## 4 Canadian Case Studies: Data Quality in Action

The following four case studies provide real-life examples of the diverse tools used by CIHI to ensure high-quality health and health system data.

Case Study 1 highlights CIHI's Continuing Care Reporting System and how its unique data standard and point-of-care data capture using electronic health records provide both the incentive and the technical support for prevention of errors.

The next two studies provide examples of CIHI data quality monitoring activities. Case Study 2 describes a one-time reabstraction study on emergency department data that identified root causes of coding errors. Case Study 3 describes a tool used in an annual assessment of hospital financial and activity data and widely reported to key stakeholders.

The final case study illustrates how the CIHI responded to a known issue with a training intervention as part of the continuous improvement process for coding of intervention data in acute care.

### 4.1 *Case Study 1: Preventing Data Quality Issues in Continuing Care*

Continuing care in hospitals and residential care facilities (such as long-term care, nursing or personal care homes) is an important component of Canada's health care system. These facilities serve populations who do not need the services of an acute care hospital but have health problems that do not allow them to live at home, even with home care support.

CIHI recently developed a new state-of-the-art reporting system to support the growing numbers of provinces and territories participating in the Continuing Care Reporting System (CCRS). The CCRS provides clinicians, managers, policymakers and the public with high-quality information on persons receiving facility-based continuing care.

Since its predecessor's launch in 1996, adoption of the CCRS data standard has expanded from 120 to more than 1,000 facilities in eight provinces and territories across Canada.

During the design phase for the new reporting system, prevention was a primary focus. Lessons learned from previous data quality evaluations and from client feedback were applied to ensure that potential errors were flagged as early as possible in the data capture process, correction processes were streamlined and data quality reporting was improved.

While CCRS data will be subject to comprehensive monitoring and continuous improvement strategies as described previously, the unique strengths of its prevention strategies are the focus of this case study.

#### **4.1.1 The Data Standard**

As discussed previously, data quality depends on a high-quality data standard. The data standard for the CCRS is the Resident Assessment Instrument Minimum Data Set, Version 2.0 (RAI-MDS 2.0), developed by interRAI, an international research network.

The assessment has undergone rigorous reliability and validity testing in a number of countries worldwide, including Canada [6, 8], and represents a solid foundation for capturing high-quality data.

The RAI-MDS 2.0 assessment captures the clinical and functional characteristics of residents, including measures of cognition, mood and behaviour, continence and skin condition, medications and resource utilization.

A unique feature of the RAI-MDS 2.0 system is the suite of decision support tools that are derived from the assessment. These were developed by interRAI using research and best practice information from around the world. The tools and their uses are briefly described in Table 4.

Clearly, the relevance and usefulness of the assessment and the outputs to both clinicians and system managers provide one of the most powerful incentives for the capture of high-quality data.

#### **4.1.2 Point-of-Care Data Collection**

The next critical component of prevention in CCRS is the approach to data capture. Ideally, clinical information is captured once, as part of the normal process of care, and used for many purposes. Electronic implementation of RAI-MDS 2.0 and submission to CIHI represent this ideal.

The clinician uses the RAI-MDS to assess a person and immediately enters the data into the person's electronic record. This allows for the generation of the real-time decision support tools that assist with care planning and delivery. In regions with interoperable health records, the assessment information is shared within the circle of care (e.g. physicians or hospital emergency department staff).

**Table 4** Decision support tools from RAI-MDS 2.0 assessment

Output	Purpose
Clinical assessment protocols	Person-level reports flag residents who may be at risk of decline or failure to improve Used at point of care by clinicians to inform care planning and set priority levels for service Aggregate data used by system managers to identify at-risk populations and plan for services
Outcome scales	Person-level reports summarize the clinical and functional status of residents Used at point of care by clinicians to measure clinical and functional status and monitor changes over time Aggregate data used by system managers to understand and plan for changing populations
Quality indicators	Organization-level measures of quality across key domains including physical and cognitive function, safety and quality of life Aggregate data used by quality leaders to drive continuous improvement efforts Used to communicate with key stakeholders through reports cards and accountability agreements
Resource utilization groups	Organization-level reports cluster populations with similar characteristics and resource use Used by system managers and provincial/territorial funders as an input to resource allocation and funding decisions

Electronic point-of-care data collection also allows for production of aggregate reports for organization and health system use. When the data flow to CIHI, organizations have access to comparative reports on quality and utilization that they can benchmark with their peers.

CIHI issues technical specifications for software vendors who build the point-of-care applications for capture and reporting of RAI-MDS 2.0. These specifications allow software vendors to build in comprehensive edit and audit rules to prevent the capture of erroneous or suspicious data. CIHI also issues detailed standards for generation of the statistical outputs, ensuring consistency of reporting across the country.

CIHI's annual vendor testing program ensures that data coming into CIHI are error-free (i.e. pass edit checks) and that they are appropriately generating the decision support tools for their users.

#### 4.1.3 Training and Support

The first and most important custodians of data quality are those who capture the information—in this case, clinicians. The RAI-MDS 2.0 manual developed by interRAI and adapted by CIHI for Canadian use is clinician-friendly, providing rationale for the assessment item, detailed process and coding directions, with real-life examples and references.

CIHI augments this manual with an innovative program of training and support on the data standard, data submission processes and the use of the decision support tools. Nurses are the primary audience for the data standard and clinical decision support training, and their needs are quite distinct from those of trained health record coders in hospitals.

Nurses working in long-term care or nursing homes are often new to computers, standardized assessments and statistical reports. Their facilities may not have a strong technical infrastructure. There may be limited funds for training. Nurses and other clinicians may have difficulty getting coverage for their care duties to attend training.

Following an evaluation and extensive consultation with clients, CIHI designed a new training program that accommodates the flexibility these clinicians need. A modular curriculum allows clients to build their own program and learn at their own pace. The power of web-based learning tools is leveraged, with interactive programs that engage learners in group discussions and information sharing.

The training also includes modules designed for analysts, clinical managers, quality leaders and system executives who are the primary users of the aggregate information.

#### **4.1.4 Results**

Participation and feedback on the new training program have been excellent; formal evaluation is underway. Through a shift away from face-to-face workshops, the new program has improved client access, increased CIHI's capacity to reach the growing client base and reduced costs for both CIHI and the participants.

It is too early to report on the performance of the new CCRS technical solution that will streamline and improve many data quality processes. What we do know is that continuing care data quality was good prior to the launch of the new system. The data have been used for front-line and system-level decisions across the country, a testimony to its quality.

With the new system and innovative approach to training, we are confident that quality will continue to improve.

### **4.2 Case Study 2: Monitoring Data Quality in Hospital Emergency Departments**

Emergency and ambulatory care services are among the largest-volume patient activities in Canada and are a key component of the continuum of health care. To better understand how this sector is evolving and performing, CIHI developed the National Ambulatory Care Reporting System (NACRS). Hospitals, health regions and health ministries receive information on visits, clinical characteristics and services provided in day surgery, outpatient clinics and emergency departments.

Client visit data are documented at the point of care, coded and abstracted by health records personnel and submitted to CIHI. CIHI provides hospitals with data quality feedback in operational reports and produces comparative statistical reports including wait times by triage level.

CIHI, in collaboration with the Canadian Health Information Management Association, conducted a study to evaluate the quality of the emergency department data in the National Ambulatory Care Reporting System (NACRS).

Detailed documentation and executive summary of the study may be found on the CIHI website [2].

#### **4.2.1 Study Design**

This data quality study had two components: a reabstraction study and a questionnaire on data quality practices.

The reabstraction study used a two-stage stratified probability sample. Facilities were the primary sampling units and the emergency department visits within selected facilities were the secondary sampling units. The study sample was drawn from the NACRS 2004–2005 database, resulting in 7,500 abstracts selected from 15 hospitals.

Information on the emergency department documentation and processes was also collected through a retrospective questionnaire sent to the participating hospitals. The survey questions targeted areas that were thought to have potential influences on data quality: staff experience and training, coding and abstracting practices, the data collection process and chart documentation and data quality initiatives and programs available in the facilities.

#### **4.2.2 Data Collection**

Health information management professionals were recruited to perform field collection for the reabstraction study. Each candidate underwent screening requirements (e.g. interviewed, trained, tested) to ensure high-quality reabstracted data. Only those persons who succeeded in these screening requirements collected data for the reabstraction study.

For field collection, reabstractors travelled to hospitals and captured data on software developed by CIHI. The software application and underlying data sets were loaded onto each reabstractor's individual laptop prior to data collection. Reabstractors reviewed the patient chart documentation, coded and abstracted the data and entered it into the application. Reabstractors were provided with a variety of coding resources and were able to contact CIHI if further assistance was required. In addition, reabstractors were able to discuss with one another any issues they encountered while coding.

Only after they had entered the data were they able to compare these against the data originally submitted to the NACRS. Discrepancies between the original and reabstracted data were automatically generated by the application and reabstractors assigned reasons to any observed discrepancies.

Data collection for the hospital questionnaire was a separate activity. The questionnaire was mailed to and completed by the health records managers within the selected hospitals. Respondents were asked to consult with hospital personnel and record information that reflected the circumstances at their hospital during the study period. The questionnaire was completed by all 15 hospitals, with a few instances of item nonresponse.

#### 4.2.3 Processing and Analysis

Records in the reabstraction study were assigned weights based on their probability of selection and to account for nonresponse. This allowed estimates from the study to be representative of the study target population.

Data collected from the hospital questionnaire were used in conjunction with reabstraction study data such that comparisons could be made between facility group estimates.

Two methods were used to divide facilities into mutually exclusive groups based on the similarity of their questionnaire responses. In most cases, hospital groups were based on the response to a single question. Otherwise, hospital groups were created using hierarchical data clustering algorithms based on responses to multiple related questions.

Point estimates were generated using the Horvitz-Thompson estimator [3], and 95 % confidence intervals were generated using the Taylor linearization method [7].

In the analysis that compared results between mutually exclusive hospital groups that were based on questionnaire responses, the goal was to determine whether certain hospital behaviours were associated with higher or lower reabstraction study agreement rates. Estimators for differences in population proportions were used to determine statistical significance [4]. However, differences observed between hospital groups only suggest that a particular hospital practice or policy affected data quality measure outcomes. Other factors, not used to group the hospitals, could also explain the differences.

#### 4.2.4 Results

The hospital questionnaire enabled the categorization of hospital groups for comparison of reabstraction study estimates. Certain organizational practices or procedures were associated with consistent differences in results.

The following table illustrates the process of grouping hospitals and generating reabstraction study results. Hospitals were asked whether they afforded time during the regular workday for coders to review the coding and abstracting standards and guidelines on an ongoing basis. Twelve indicated that they did afford this time and three indicated that they did not. Hospital groups were created based on this response and reabstraction study agreement rates were generated (results for two data elements are shown in Table 5).

**Table 5** Exact code match agreement rates in the reabstraction study for different facility groups

	Main problem (%)	Reason for visit (%)
Facilities that afford time during the workday	$78.9 \pm 1.4$	$68.1 \pm 1.6$
Facilities that <i>do not</i> afford time during the workday	$76.5 \pm 2.9$	$57.7 \pm 3.4$
Difference statistically significant ( $p \leq 0.05$ )	No	Yes

Agreement rates were higher for the hospital group that afforded the time to review the coding standards. Significant differences between hospital groups were found for the code assignment for Reason for Visit.

Similar analysis was repeated for other hospital groups created from the questionnaire responses. Coding and abstracting practices, the tools that promote accuracy and consistency in coding, varied across the facilities. Higher agreement rates for coding were observed in facilities that:

- Afford time during work hours for coders to review the coding standards
- Have dedicated coders to abstract emergency department visits
- Develop facility-specific guidelines to complement the directives provided by CIHI
- Provide clinical documentation beyond the emergency department record form for the health record to be considered “complete”
- Involve the administrative team in documenting the patient’s triage
- Always comply with the CIHI national standards
- Have standard procedures that detail documentation requirements
- Provide formal physician training on chart documentation requirements

The findings from the study illustrate how the understanding of data quality issues is enriched by employing more than one study method. In this instance, the information gleaned from the reabstraction study was enhanced with a hospital questionnaire.

The study provided this program’s key stakeholders with critical information which served as a foundation for ongoing improvements and a baseline for future studies to document progress.

#### **4.3 Case Study 3: Monitoring Financial and Statistical Data Quality in Hospitals**

The Canadian MIS Database (CMDB) contains financial and statistical information from hospitals and health regions across Canada. The data are collected according to a standardized framework for collecting and reporting financial and statistical data on the day-to-day operations of health service organizations.

The framework is known as the Standards for Management Information Systems in Canadian Health Service Organizations (MIS Standards). Data are submitted to CIHI by provincial/territorial governments.

Currently, most information in the CMDB is specific to hospitals. In provinces and territories where hospitals are part of a regional health authority, regional data are also submitted to the CMDB, providing a more complete picture of health services for that region.

Financial and activity data are used to develop key indicators by hospital and functional centre. These include administrative expense indicators as well as cost and worked hours per weighted case, allowing for comparisons of efficiency across peer organizations and jurisdictions.

#### 4.3.1 Methods

The MIS Compliance Assessment tool is used to assess the quality of a jurisdiction's data submission on an annual basis. This tool was developed after extensive consultation with representatives from each reporting jurisdiction.

The methodology is applied specifically to hospital data provided to the CMDB and asks five questions of each provincial and territorial submission:

1. Where required, has the province or territory signed off on a table to map provincial or territorial primary and secondary accounts to the MIS chart of accounts? If so, this dimension is scored at 100 %. If not, the jurisdiction is assigned a score of 0 % and the data are considered non-compliant.
2. Was the submission received by the submission due date? If so, this dimension is scored at 100 %. If the submission was late, the score for this dimension is discounted with compounding “interest”.
3. Does the submission use the MIS minimum chart of accounts? The score for this criterion is the percentage of expenditures/revenues reported in minimum CMDB primary and secondary accounts. By weighting the scores, small functional centres or providers have less impact on the overall score than larger functional centres/hospitals.
4. Does the submission include the reporting of minimum statistics required in the functional centre framework? For each statistic required the methodology measures the percentage of expenditures that ought to have a statistic that do have the statistic. The overall score across required statistics is weighted by functional centre expenditures so that statistics required in a few or smaller functions have less impact on compliance than statistics that are broadly required.
5. Do the financial and statistical data appear reasonable? The last and most rigorous test looks at whether the data being reported are reasonable. This assessment is based on an iterative multivariate analysis of multiple variables for each patient care functional centre controlling for cohort (small, community, teaching) and provincial/territorial labour rates. Extreme outliers ( $p = 0.01$ ) are identified at each iteration and flagged as failing the test of reasonableness. The test concludes when no additional outliers are flagged and the score is based on the percentage of expenditures that pass each assessment.

The overall score is the product of each of the component scores and can be roughly interpreted as the percentage of resources that pass all data quality tests. It should be noted that data failing a test in one dimension are not assessed in subsequent dimensions.

#### **4.3.2 The Feedback Process**

The compliance tool changed the annual production cycle of the Canadian MIS Database. Shortly after the submission of data to the CMDB by the jurisdictions, preliminary assessments are shared with each jurisdiction. CIHI staff then meets with representatives from each jurisdiction to discuss the results and areas for improvement. The jurisdiction is then provided a grace period to resubmit corrected data to the CMDB. After this period has ended, the data are generally considered to be final.

In addition to assessing and scoring each submission, the methodology produces detailed diagnostic reports that are used to:

- Illustrate all examples of non-compliant reporting
- Identify issues with the provincial or territorial mapping tables that may be corrected
- Identify individual providers that may require additional support to improve CMDB reporting
- Identify issues that may be common across providers and help prioritize data quality initiatives to improve future submissions

These reports feed into CIHI's continuous improvement processes and guide future enhancements to the data and technical standards, training, client support and reporting.

#### **4.3.3 Results**

The MIS Compliance Assessment tool has been informing CIHI data quality reports for data suppliers and Deputy Ministers since 2006.

The assessment tool has proven to be extremely flexible; not only can it be easily adjusted to reflect changes to the MIS Standards, but it also can be expanded to review data from outside of the hospital sector. In fact, CIHI has developed an expanded MIS Compliance Assessment to adjudicate the quality of MIS data in all sectors. This expanded assessment will begin to inform CIHI's Deputy Ministers' Reports in 2013.

Over the 6 years that the tool has been used, the quality and compliance of virtually all jurisdictions' data has improved. Persistent MIS data quality issues such as the use of clearing accounts and invalid functional centres have decreased in both frequency and materiality. Further, the reporting of statistics has improved, particularly in smaller jurisdictions.

Other benefits from the implementation of the assessment tool are less tangible but just as important. The results of the assessments are provided to all participating jurisdictions, facilitating conversations and sharing of successful MIS reporting practices across Canada.

## ***4.4 Case Study 4: Resolving Acute Care Coding Issues with e-Learning***

In a needs-assessment survey conducted in 2009–2010, CIHI clients in the acute care sector rated the coding of flaps and grafts interventions as a high priority for CIHI education.

Flaps and grafts are methods by which a wound can be closed. A flap refers to tissue used for reconstruction or wound closure that retains all or part of its original blood supply after the tissue has been moved to the recipient location, whereas a graft is a tissue of epidermis and varying amounts of dermis that is detached from its own blood supply and placed in a new area with a new blood supply.

Intervention coding in Canada is captured using the Canadian Classification for Health Interventions (CCI), which is maintained by the Classifications and Terminologies department at CIHI.

Following the survey, an e-learning course to address these coding issues was developed and launched the following year (April 2010). The effectiveness of this course was evaluated to assess the impact on coder knowledge and confidence and coding consistency.

### **4.4.1 Study Design and Data Collection**

A reabstraction study in one Canadian hospital was used to assess the impact of the education on intervention coding consistency. Coders participating in the study also completed a questionnaire before and after they completed the e-learning course to self-rate their knowledge of the standards and confidence in coding flaps and grafts.

Inpatient and surgical day care abstracts with CCI interventions of the skin and soft tissue with a flap or graft tissue qualifier (the 10th digit of the CCI code) were the focus of the study. Only the operative reports were reviewed and the data collected during the chart review were compared with the data previously collected by the hospital.

The study had two data collection stages: the pre-education phase and the post-education phase, differentiated by the date when coders at the hospital participated in the e-learning course “Coding Flaps and Grafts of Skin and Soft Tissue”.

The education was completed in June 2010 and was timed so that it occurred after the coders had finished coding the 2009–2010 abstracts but before they began coding the 2010–2011 data.

Inpatient and surgical day care abstracts were sampled in the two phases based on abstract discharge date and reabstracted by a CIHI Classifications Specialist:

- Pre-education phase: 124 inpatient and 50 surgical day care operative reports with discharges from April 1 to September 30, 2009.
- Post-education phase: 133 inpatient and 47 surgical day care operative reports with discharges from April 1 to September 30, 2010.

#### 4.4.2 Results

A significant improvement was noted in coding consistency of local flaps in the inpatient data following completion of the course. Of the local flaps abstracted during the pre-learning phase, 61 % had no local flap confirmed in the reabstraction chart review. In the post-learning phase this variance decreased to 25 %.

There was a slight improvement in coding consistency of free flaps pre- and post-education (84 % and 89 %). Coding consistency for split-thickness grafts remained the same in both phases of the study (89 % and 88 %). There were insufficient numbers of interventions studied with a full-thickness graft and a pedicle flap to assess the quality of coding consistency.

There was high agreement on coding local flaps in the day surgery data in both the pre-education and post-education phases (95 % and 97 %). An improvement was seen with the coding consistency of split-thickness grafts following completion of the course (75 % pre-education and 100 % post-education). There were insufficient numbers of interventions studied with a full-thickness graft, free flap and pedicle flap to assess the quality of coding consistency.

In the post-education questionnaire, hospital coders reported being more knowledgeable and more confident in their ability to select the correct CCI qualifier to describe the flap and graft closure performed. The study suggests there is statistical reliability between the learners' self-assessments and the change in their coding consistency.

Overall, the e-learning course "Coding Flaps and Grafts of Skin and Soft Tissue" increased the knowledge, confidence and coding consistency of hospital coders, demonstrating the effectiveness of this training strategy for improving the quality of abstracted data.

## 5 Opportunities for the Future

While there are many challenges facing today's health information professionals, two opportunities in particular will help to address current challenges and have an important impact for CIHI and its key stakeholders. These are the evolution of interoperable electronic health records and the movement toward integration of clinical, financial, human resource and other administrative information across the continuum of care.

## **5.1 *Electronic Health Records***

In Canada, as well as elsewhere in the world, efforts are underway to develop electronic health records that transmit and store a person's health information for access and use by authorized personnel and to provide system managers with aggregate information. The demand for more and more timely data will continue to grow, as will the need to manage the data collection burden on the front lines.

These developments will transform the way data are captured and used, requiring adaptations along the data supply chain.

Progress has been made and lessons learned. Since 1996 CIHI has been receiving clinical data captured electronically at the point of care in continuing care hospitals. In recent years, the home care, mental health and rehabilitation sectors have reaped the benefits of this data capture strategy. CIHI is also beginning to receive primary care data through electronic medical records in physician offices and clinics.

The CCRS case study illustrates the potential for harnessing the power of electronic health records and highlights selected strategies for managing data quality in this environment.

CIHI will continue to work with key stakeholders to ensure that the development of electronic health records incorporates the needs of both primary front-line users but also the needs of those who use the aggregated information for system planning, quality measurement and accountability.

## **5.2 *Data Integration***

Another key development, in part related to the trend toward integrated health service delivery, is the movement away from health information "silos". This presents unique challenges and opportunities for CIHI as well as health system data providers.

Information systems at CIHI and in the field have evolved over time, with hospital systems emerging first in response to information needs related to the Canada Health Act. In recent decades there has been growing interest in sectors beyond hospitals, such as primary care and home care.

Data providers and users want seamless information on individuals who access services across the continuum. There is also a need to integrate clinical data with financial, human resource and other administrative data to evaluate efficiency and identify cost-effective interventions or programs.

Data standards have been developed over the years in consultation with sector-specific stakeholders with diverse needs and interests. Therefore, there are differences in data standards across systems that limit the usefulness and comparability—and therefore the quality—of the data.

CIHI is working toward improvements in data and report integration, conducting ongoing consultations with stakeholders including key standards and privacy

organizations. Continuing development of a CIHI conceptual data model and data dictionary inform the design of CIHI reporting systems and move them toward greater alignment.

Only when optimal data integration is achieved can we address the most complex issues in health care, such as delivering the right care in the right place at the right time for all Canadians.

**Acknowledgments** The authors wish to thank Jean-Marie Berthelot, Douglas Yeo, Maureen Kelly and Anyk Glussich for their assistance in the preparation of this chapter.

## References

1. Canadian Institute for Health Information (2009) The CIHI data quality framework. CIHI, Ottawa
2. Canadian Institute for Health Information (2007) CIHI Data Quality Study of Ontario Emergency Department Visits for Fiscal Year 2004–2005. CIHI, Ottawa
3. Cochrane EG (1977) Sampling techniques. Wiley, Hoboken
4. Devore JL (1995) Probability and statistics for engineering and statistics. Duxbury Press, Pacific Grove
5. Health Canada (2) (2010) Canada's Health Care System (Medicare). <http://www.hc-sc.gc.ca/hcs-sss/medi-assur/index-eng.php>. Accessed 18 May 2012
6. Hirdes JP et al (2008) Reliability of the interRAI suite of assessment instruments: a 12-country study of an integrated health information system. BMC Health Serv Res 8:27
7. Lohr SL (1999) Sampling: design and analysis. Duxbury Press, Pacific Grove
8. Morris JN, Jones RN, Fries BE, Hirdes JP (2004) Convergent validity of minimum data set-based performance quality indicators in postacute care settings. Am J Med Qual 19:242–247

# Shell's Global Data Quality Journey

Ken Self

**Abstract** The importance of high-quality data has been long recognised in Shell. During the 1990s, a major data management programme developed and implemented sound data management practices that were successfully implemented in a number of operating units (OUs). However, each OU adapted the tools and techniques to their own environment. A unified global approach to data quality remained elusive until the new millennium. This chapter describes Shell's global data quality journey since the early part of the millennium to the present.<sup>1</sup>

## 1 Introduction

Shell is a global group of energy and petrochemical companies with around 90,000 employees in more than 80 countries and territories (<http://www.shell.com/home/content/aboutshell>). Shell has three businesses:

- Upstream explores for and extracts crude oil and natural gas
- Downstream refines, supplies, trades and ships crude worldwide; manufactures and markets a range of products; and produces petrochemicals for industrial customers
- Projects and Technology manages delivery of Shell's major projects and drives the research and innovation to create technology solutions

---

<sup>1</sup>This chapter is the work of the author and does not necessarily represent the opinions of Shell. The information, data and opinions in this chapter are of a general nature only and should not be acted on or relied on without independent professional advice.

K. Self (✉)

The Shell Company of Australia, 8 Redfern Road, Hawthorn East, Melbourne, Victoria, Australia  
e-mail: [Ken.Self@shell.com](mailto:Ken.Self@shell.com)

The word “Shell” first appeared in 1891, as the trade mark for kerosene being shipped to the Far East by Marcus Samuel and Company. The word was elevated to corporate status in 1897, when Samuel formed the Shell Transport and Trading Company.

When the Royal Dutch Petroleum Company and Shell Transport and Trading merged in 1907, it was the latter’s brand name and symbol which then became the short form name (“Shell”) and the visible emblem (the “Pecten”) of the new Royal Dutch/Shell Group.

In 2005, the Group underwent a major structural reorganisation as the near-century-old partnership between Royal Dutch and Shell Transport and Trading was dissolved and one company was created, Royal Dutch Shell. The headquarters of the new company are in The Hague. July 5, 2007 marked the first centenary of the original partnership.

The importance of high-quality data has been long recognised in Shell. During the 1990s a major data management programme developed and implemented sound data management practices that were successfully implemented in a number of operating units (OUs). However, each OU adapted the tools and techniques to their own environment. A unified global approach to data quality remained elusive until the new millennium. This chapter describes Shell’s global data quality journey since the early part of the millennium to the present.

## 2 The Data Quality Challenge

Put simply, the greatest challenge in making progress on data quality is getting enough of the “right” people pulling in the same direction. Business managers understand the importance of high-quality information but may not have the knowledge of how to achieve it. However, to get their commitment to invest in solving the data quality issue, they need some degree of confidence in the outcome.

Consider what you or anyone would want to know before making an investment like building a house. You’d want the builder or architect to:

- Understand your goals, e.g. stay warm in winter and cool in summer, and adapt to them
- Tell you the steps you will go through to get your dream home
- Explain how those steps will result in you getting what you want
- Have a track record of successful results
- Estimate how long it will take and what it will cost
- Have enough resources to draw on to finish in time
- Show that those resources are properly qualified for the task

A leader of a business unit manages their business performance through a set of controls that manage risks. Changing those controls introduces risk to performance that any leader with accountability would rightly challenge. To bring about the

change it is necessary to demonstrate that the risk is managed as well or better than with current practices.

On a small scale, let's look at a simple case of a customer delivery address. A change to the process for maintaining that address can impact the ability for drivers to navigate to it. Changes could include who maintains the address in what format or where it is stored. The leader needs to be satisfied that the outcome of the proposed process change does not increase the risk of a driver being unable to navigate.

Now imagine making changes across a large organisation. Just getting a change proposal onto the table requires a pool of competent and experienced people who not only understand data quality principles but can also clearly communicate all aspects of what they do to a lay person.

That communication needs to address the concerns of the many business leaders who are impacted by the change. This means understanding not just the principles of managing data quality but also the specific business impacts of the changes in managing data quality – negative and positive.

Business leaders understand processes. Describing data quality in terms of the process for managing it and its impact on business processes is critical to communicating and gaining acceptance of the changes. It is the difference between explaining how “you won't have duplicate customer records” and “you will correctly price the sales to your customers”.

A characteristic of data professionals is that they focus strongly on definitions. In this chapter I do not use strictly defined terms, in line with the reality of the business environment. Information and data are used interchangeably so do not read anything into which term or the other is used. If pressed, I would say that “information” is simply “high-quality data”. Similarly the term MRD (master reference data) is also used quite widely in Shell and is sometimes interpreted as “master **and** reference data”. In this chapter I make no distinction between master data, reference data and MRD. However, the term MDM (master data management) is used to denote the tools aimed specifically at managing MRD as opposed to the process of “master data management” or derivations like “managing master data”.

### 3 Where Are We Today?

Most of Shell's MRD is managed by a global team in Finance, located in business service centres around the globe. They are trained in continuous improvement and data quality and also the business processes and the data objects they work with. Process owners have defined global standard business processes, including the data requirements for their processes, and designed MRD processes integrated with their business processes.

### 3.1 Processes

Process management is established widely across Shell businesses and functions. Global standard business processes have defined data requirements and MRD processes are integrated with them. Below a series of examples are presented that relate business processes and respective MRD:

- Offer to cash involves negotiating a sales agreement with a customer through making sales and receiving payments for those sales. The process includes maintenance of customer MRD.
- Product life cycle management involves the development and production of products for sale (e.g. lubricants) and includes maintenance of product MRD.
- Requisition to pay encompasses all activities to do with the purchase of goods and services including maintenance of vendor, materials and service master data.
- Maintenance integrity and execution encompasses all the activities related to the maintenance of plant and equipment and includes maintenance of functional location and equipment MRD.
- Hydrocarbon management is the movement of products and stock accounting and includes maintenance of storage location MRD.
- Record to report includes financial accounting and reporting of management information and statutory reporting. It includes maintenance of general ledger accounts and cost centres.
- Fixed asset accounting involves accounting for capital expenditure on assets and includes maintenance of fixed assets and projects MRD.
- Retail network planning involves determining where to best locate retail sites and acquiring, divesting or enhancing sites accordingly. These activities include maintenance of site and equipment MRD.

Management of MRD is itself recognised as an end-to-end process and needs to be managed accordingly. At the core of the end-to-end MRD process are the MRD maintenance activities. These begin when a data customer has a need for MRD to be stored in the database up until the time the MRD is ready to be transacted against. MRD maintenance activities typically include the following: submit request to maintain MRD, maintain MRD in enterprise resource planning (ERP) systems and receive and validate requests to maintain MRD.

Many requests to maintain MRD consist of updates to a large number of records, known as “bulk updates”. These are an important class of MRD activity due to their impact on data quality and process efficiency.

These maintenance activities are not sufficient to deliver the high-quality data required by the business processes. Additional management activities are needed to measure and report MRD quality, improve MRD processes and clean the MRD lake.

These are supported by a system of process and data designs, role definitions, data quality standards (DQS) and controls. Process designs describe in detail how the standard process is performed using process hierarchy diagrams, swim-lane diagrams, standard operating procedures and work instructions. Role definitions

Finance: Asset		Completeness	Accuracy
Ref	Data Quality Objective	Metric	
1	High value asset existence is recorded and the minimum required information held for GSAP is correct.	Physically, right asset and DQS asset master record details for all assets with an acquisition value greater than \$5K USD. Estimated effort is for SOX remediation.	
2	Asset Class (categorised by differing taxation treatments within a country) must be valid and accurate	The Asset Class of each asset must be appropriate for the type of asset as determined by capex/opex policy and verified during asset verification exercise.	
3	Verification date and verifier recorded for all assets	The most recently available date of actual verification of an asset, and a record of who verified it, should be held against all assets.	
4	All Assets must have a valid and accurate Asset Description (non-blank, unambiguous and consistent)	Asset description must give a meaningful description of the asset so as to enable its unambiguous identification by an unfamiliar verifier.	
5	Low value assets are not held in the Group book asset register	Assets at or below \$5K USD acquisition value (or \$1K USD with Controllers dispensation for a country), but above \$0, in Group book are to be written off. Tax/Stat assets must be held if required for local legal/fiscal needs.	
6	All Assets must have a valid and accurate Plant & Location.	All assets must correctly map to a single to-be Plant/Location (non-blank, non-corporate).	
7	All asset descriptions are to be recorded in a common shared services supported language.	Asset description must be maintained with a meaningful English or other supported shared services language description such that a non-local verifier can read and locate the asset. A mixture of languages must not occur within the one register.	
8	Room field for Retail assets and B2B assets located on customer sites	Retail site assets should store the Retail Site identifier (MRN) in the room field. For B2B assets which are located on customer sites, the Room is the Customer's current contract ID.	

*Clarity/Uniqueness*      *Relevance*      *Consistency*

**Fig. 1** An example of data quality standards (DQS)

identify all the players in a process and the activities they perform at an atomic level that enables them to be aggregated into jobs. Data designs describe the standard meanings, formats and business rules for data. DQS identify key characteristics of each data object that are critical to business objectives and procedures for measuring their compliance to standards. Key controls identify major business risks that require mitigations to be embedded in the processes such as access controls, segregations of duties and necessary approvals.

Reports are produced regularly on a number of process metrics including data quality, maintenance request turnaround time, invalid requests and controls compliance. Data quality is measured by compliance with DQS for all key data objects stored in databases. This is expanding to include measuring the compliance of maintenance requests against the same standards. Most of the compliance checks on the databases are automated, but where this is not possible, samples from the database are checked manually.

A typical set of DQS is shown in Fig. 1. These consist of an objective which states the data outcome that is required and which aligns to a desired business outcome. The metric describes more specifically how data quality is measured. The DQS cover a range of data quality characteristics.

Continuous improvement is established for MRD processes using Lean Sigma methodology [1]. Process issues are identified using metrics and feedback from stakeholders. Projects to address those issues are prioritised according to the benefits with priority going to projects with bottom-line benefits over those with softer financial benefits or nonfinancial benefits. More and more benefits in the business processes are being identified as estimates of the cost of poor quality data improve.

### 3.2 Technology

Compared to many other companies, Shell's approach to master data management appears "low tech". This is due primarily to the strong emphasis placed on establishing the end-to-end process and organisation. The catch phrase is abbreviated to ESSA: eliminate, simplify, standardise and automate. That is, eliminate unnecessary activities and simplify the processes, then implement a standard process across all domains and then automate the standardised and simplified process.

A large-scale integrated MDM solution is not yet in the solution set. A variety of tools have been implemented to provide equivalent functionality.

Instead of a central master data repository, a small number of global ERPs act as the MRD repository for applications connected to them. Those ERPs are aligned to key businesses and functions. The level of interaction between these ERPs and sharing of data is low.

Workflow is used extensively in MRD processes. Some processes use purpose-built workflow applications but others use more rudimentary tools: spreadsheet forms sent by email and managed in document management tools for tracking. Data input on request forms is minimised and inputs are validated on the form. Workflow routing ensures compliance with controls and data is gathered on the work flow either automatically or manually to report on performance.

A Data Quality Tool (DQT) is used to collect MRD data weekly, run quality reports and produce reports on defective records and data items. The latest version enables the QA (Quality Assurance) staff to improve the queries as the business rules evolve and reduced reliance on IT specialists. Most of the Quality Checking (QC) is automated but some checks are still performed manually. Much of the QC is still focused on data in the repositories but the DQT is being used increasingly to report on the quality of data coming through the update streams. This is possible with an automated system because it checks every record and can therefore compare with the previous reporting period to detect changes that either increase or decrease compliance.

The bulk update tool enables data maintainers to extract existing data into a spreadsheet, apply updates then bulk load with each record undergoing normal system validation. The validators also apply nonsystem validation rules manually or encode them into the spreadsheet.

### 3.3 People

People are the most important element in data quality and receive the strongest focus in Shell's data quality programme. There are three broad groups of people identified with a role in data quality:

- Data practitioners who work full time managing data quality primarily in validating and maintaining MRD in ERP systems.

- Data customers who are dependent on data to perform their jobs. Simply put, that is just about everyone in the company.
- Business process owners and managers who contribute to the end-to-end activities but whose primary focus is on executing and improving business processes.

Some 80 % of MRD, measured by effort, is managed globally by the MRD team in Finance. Most data practitioners are in business service centres in key locations around the world. They manage a wide range of key master data objects used across all business processes in all business and functions in all countries and in all global ERP systems. They are responsible for the end-to-end activities across the MRD life cycle. There are also significant parts of MRD managed by global and regional organisations, but this case study focuses on the Finance-based organisation.

Process owners have defined global standard business processes. They have defined the data requirements for their processes and designed MRD processes integrated with their business processes. They receive reports on data quality and drive process improvement projects.

Clearly defined process roles and data roles enable optimal separation of activities between business process performers. Process managers work with the MRD team to identify activities that can be migrated beneficially from business process performer to the MRD team. Migrations reduce the involvement of process performers in MRD life cycle activities and allow them to spend more time on the business process. The process staff gather data request and approve creation, update and deletion of MRD and provide requirements for and feedback on data quality and MRD process performance.

Our data practitioners in service centres are trained in continuous improvement using Lean Sigma and data quality training builds on Lean Sigma training. They are also trained to be proficient in the business processes and the data objects they work with. All training includes methods to assess the proficiency of staff in the subject. This enables us to identify individual competence gaps, readiness for further development and to prioritise training delivery. Basic training is available for business process staff to explain the role that they play in ensuring data quality.

## 4 First Steps (2004–2006)

Shell's Downstream business began a major business process re-engineering initiative called "Streamline" in 2002 with the goal of implementing global standard processes across the business and drive significant cost reductions. This included the migration of over 100 ERP systems to a single global ERP. Process teams led by a process owner were established to simplify and standardise business processes in all operating units. A programme office worked with the process teams to standardise the design methodology and documentation across all teams.

From the outset, master data was recognised as critical to process success. A small MRD team was established in 2004 to complement the work of the process

teams and provide a consistent approach to the design and implementation of master data management. Even though the importance of master data was acknowledged, it was essential to build a strong case for action for significant improvements to be implemented.

Quantifying the “size of the prize” was the first step. Each process team was developing the business case for simplifying and standardising their business processes. To make the case for changes to master data management, each process team identified the dependency of their process standardisation benefits on having high-quality, standardised data. The MRD team consolidated these into a consolidated business case for master data.

A business case needs to balance the value of change against the expected cost of that change. That in turn requires quantifying the size of the problem, understanding the nature of the problem, proposing a possible solution and costing the solution.

A maturity assessment of master data management showed that parts of the business were further up the maturity curve than others, some arguably best practice. However, considering the business from a global perspective, it was “fragmented”. Fragmented aptly describes the situation where governance is unclear. Either many owners working at crossed purposes or no accountability resulting in inaction on issues and management of data is seen as a business cost. These lead to poor data quality and consequent business issues:

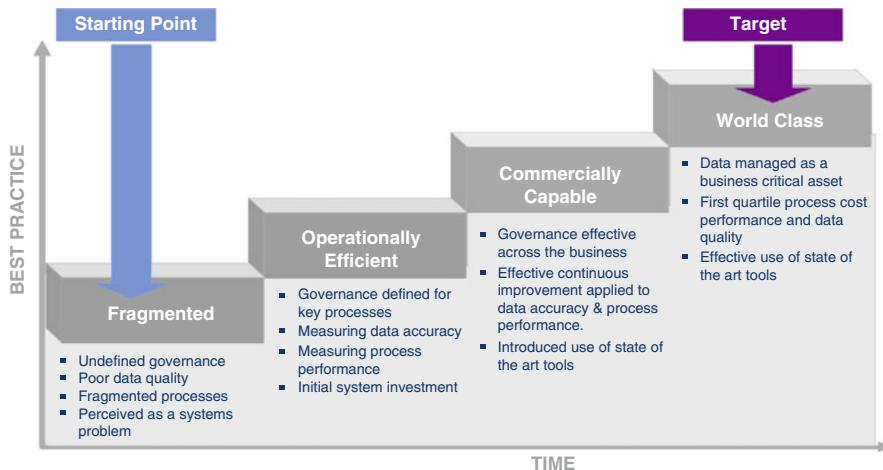
- Processes: There was no consistent approach to data management across the organisation. Each country and business had its own terminology. All working to locally defined processes and standards.
- Systems: With hundreds of ERP instances and thousands of other applications there was no clear picture of what data was required and where it was stored. So there was no consistent view of the business as every system has its own view of the data, and quality was unknown.
- People: Data management activities were fragmented across the organisation, accounting for a small percentage of many people’s daily activities.

Moving up the maturity scale requires establishing clear governance over data, implementing standard processes in key areas, defining DQS and measuring data quality and process performance. The cost of managing data is still a focus but operational management of data quality is accepted as critical to the success of business processes.

Figure 2 shows a maturity model for master data management that seeks to present the maturity levels in a “business-aligned” language.

A maturity model is a useful tool for an enterprise to establish where it sits relative to its peers and identify where it wants to be in the future.

There is any number of versions of data management maturity models, but they all have a common theme: You cannot leap from having poor data management to having great data management in one go. There are a number of stages you need to go through, and you have to consolidate on each stage before moving on to the next.



**Fig. 2** Master data maturity target

#### 4.1 Processes

Governance of master data was integrated with the governance already established for business processes. Process owners were accountable for designing the master data processes specific to objects critical to their business process. The data process design also encompassed the data definitions. For example, the process owner for Product Lifecycle Management was accountable for the process for managing product master data and for the definitions of product master data. Accountability for data content remained with the business and was re-emphasised at every opportunity.

Process information architects were established in each process team to assist the process owners with managing the simplification and standardisation of the data processes and definitions and ensuring they were documented fully and consistently. They were data experts from different lines of business and countries with backgrounds in different ERPs and architectures. Establishing best practice requires collaboration to review the diverse practices and develop ideas by distilling the best elements, test them against business requirements and develop a consistent end-to-end design. The MRD team coordinated the work of the process information architects to ensure a consistent approach to managing master data across all process teams and data objects.

The MRD processes were designed to enable MRD maintenance to be separated from other business activities to address the problem of inexperienced part-timers managing master data. The end-to-end master data process begins with the requirement for data to be stored and ends when the master data is ready for transacting business. “Requestors” ask for MRD to be created or updated; “approvers” in the business ensure that the request represents a valid business needs;

“validators” check that the request conforms to the data design; “maintainers” create or update MRD in the ERP. These roles do not assume any organisational structure as an individual could hold both requestor and validator roles, subject to business control requirements. However, it enables the “disentangling” of MRD and business activities where it is beneficial.

It is said that “you can’t manage what you don’t measure”; so managing master data quality requires a way to measure it using DQS and a measurement process. The process information architects focused on the fifteen most important master data objects and defined, for the first time ever, the “top 10” critical quality requirements for each one against which we could measure, and therefore manage, data quality. The DQS are supported by QC procedures which are clear and detailed work instructions to validate the data against the DQS.

The initial target for QC was the legacy data cleansed by operating units for migration to the global ERP system. The cleanse effort was estimated as hundreds of man-years that needed to be managed to ensure realisation of the business benefits of standardisation. Normal practice on other systems implementations relied heavily on trial conversions to ensure the validity of the data and some sort of business review and sign-off on the data. The DQS and QC procedures would provide a more objective method for managing the data cleanse. An objective for QC was that it should not take more than 5 % of total cleanse effort. Due to the large volume of data to be checked a sampling approach was employed to provide input to the QC procedures. A random sample of 200 records was sufficient to provide an accurate estimate of data quality.

It is not sufficient to measure only data quality. Process performance measures are also needed to improve operational performance. Additional measures included the end-to-end process time, turnaround time for performing updates and accuracy of requests received.

## 4.2 *Technology*

MDM solutions were becoming more established in the marketplace in 2004. In our investigation into implementing an MDM solution alongside the global ERP, it was evident that these tools were not yet mature. They did not support the breadth of data objects, they did not include range of functionality required and considerable development effort was needed to implement them. On the other hand, the global ERP could itself fill the role of a central master data repository and other functionality could be implemented using established technologies.

Workflow was recognised as an important tool for master data management, especially when large volumes of updates need to be managed. Data is typically sourced from a number of different roles, each contributing their part of the record. One or more business approvals are typically required to maintain master data. Workflow supports the timely approval and processing of update requests and reporting of performance.

### 4.3 *People*

An important finding came from a survey of master data management practices in different operating units. The survey found that over 2,000 people had update access to master data in ERPs but were managing master data mostly on a part-time basis. The average MRD worker spent 20 % of their time on master data management processes, and the rest was on business processes. Some spent as little as 1 % of their time on master data management. All were working to locally defined processes and standards. Most had minimal training on their activities and low levels of experience and were unfamiliar with business rules and principles of quality management.

The approach for managing master data centred on consolidating the fragmented activities into full-time master data management professionals. Fewer people having access to update master data gives immediate benefits in quality by reducing the variability in the process. It is easier to train fewer people on standardised processes, and full-timers are able to further learn the processes and their nuances in depth and improve them.

In 2004 Shell was establishing service centres for processing financial transactions. The service centres were set up to deliver continuity of services with fully documented processes operated by trained professional staff and managed by performance metrics and assured controls. The service centres were an ideal vehicle for managing master data. Master data activities performed part-time by business staff would be migrated to full-time service centre staff who were trained in the newly standardised processes.

Data managers aligned to the business processes were established in the MRD team to manage the migration of work to the service centres and the operational performance of master data processes performed by the service centres. The data managers were the interface between the operating units and the service centres.

The master data service centre was first tasked with performing QC on data migrations until the first operating units went live with the newly standardised business processes and global ERP. Operating unit resources were engaged in preparation for implementing the new processes and cleansing legacy data and had no capacity for QC. The MRD staff were also able to perform the checking independently and consistently and without any bias towards legacy business rules or perceptions of legacy data quality. They would also gain familiarity with the standards in preparation for the migration of operational work.

The first results of QC presented a significant change management challenge to both the operating units and the newly formed MRD team as the measured quality was considerably worse than expected. Quality results in the order of 60 % were well below the expected 80 %. So the questions were whether the standards were too strict, were they being measured properly, was the sampling method statistically sound, or was the data really that bad? It became clear that the implications of global standard processes and data would need to be communicated carefully to every operating unit during their migration preparations.

A further challenge came with the migration of work to the service centres. Managers in the operating units were concerned about the impact on their business of having master data managed by people outside their direct control. Data managers engaged the operating units to explain the business case and gain their commitment to the migration of work. This was aided by the level of documentation and training provided to the service centres, the data quality results measured in each operating unit and the continuity of service and significant cost advantages of the service centres.

## 5 Establishing Operational Capability (2006–2008)

The next step on the maturity model was to become operationally efficient through consistent execution of the MRD processes. Key to this was ongoing measurement of data quality and MRD process performance against targets. Also, the expectation from the business was that the MRD team, as full-time practitioners, would provide strong support and guidance to business units on the MRD processes.

### 5.1 *Processes*

By 2006 the first operating units had gone live with global standard business processes supported by the global ERP system. For all of them this represented a huge change to the way they worked. Even with rigorous testing of the processes and comprehensive training, the challenge of operating in a live business was considerable.

The MRD team was also performing operational processes for the first time. With this change an additional focus for master data management was the timeliness for processing requests. This was measured as both the end-to-end time and the service centre turnaround time. The end-to-end time was measured from the time a request was first created in workflow until the master was ready to have transactions processed against it. The service centre turnaround time measured from the time the service centre received the request, after all business contributions to the data and approvals were complete, until the master data was ready for transactions.

The end-to-end timeliness measures showed large variations both across and within operating units. In some operating units, all requests were completing within target, in others most were within target with a significant number substandard, and a third group was recording nearly all requests as substandard. A project team with experts from the business process team, MRD team and operating unit was supported by the Streamline programme office to apply continuous improvement methodology to resolve the problem. The project team met in a series of workshops to walk through the process comparing how the process was being performed against the design to identify the blockages. The causes of the issues they identified

included approver roles that were not assigned, incorrect measurements, attempts to follow the old processes and a higher than expected incidence of “bulk requests”. By systematically addressing the issues process by process and operating unit by operating unit, the targeted performance was achieved in most cases by the end of 2006.

The MRD team also began the first operational data quality checks on a monthly basis, using the same sampling method that was used to check the cleanse progress in the operating units. There was also a considerable task ahead to complete the cleanse work as the first live checks gave results below target. The monthly checks were time-consuming but were invaluable for tracking the progress of the live cleanse effort and for garnering support for the work from the operating units.

## 5.2 *Technology*

For many data objects, the volumes of requests did not justify a full workflow solution. Requests were submitted on spreadsheets with high rates of rejection due to incorrect filling in. The MRD team developed “Smart Forms” which are spreadsheets containing only the fields required and with drop-down lists and in-built validation to prevent errors.

The number of “bulk requests” created a need for a bulk-loading tool. In many cases these bulk requests were normal business practice such as updates to vendor catalogues or equipment inventories produced by capital projects. The volume of live cleanse work would also benefit from a bulk-loading tool. The selected tool was a relatively simple after-market tool added on to the global ERP. It enabled a selection of records to be extracted into a spreadsheet for update and applies the updates through a standard interface to the ERP that ensures normal validation of the data.

The need for automated QC was critical to reducing the high effort to perform monthly QC. Sampling was suitable for reporting on data quality but did not support remediation of the data as the defective records could not be identified. The DQT needed to report on a large range of data objects, across a range of systems and with diverse business rules. Initially implemented for customer and product master data, the DQT supports large-scale cleansing and reduces the effort needed to carry out remediation. It checks the full population of data to identify all defective records. With the sampling approach, further analysis is required to identify all non-compliant records based on the sample results. It reports the specific business rules in error on a weekly reporting cycle that enables better tracking of cleanse progress.

The subsequent roll-out of the DQT to legacy systems has enabled alignment of legacy standards to the new global standards which reduces the cost of moving to global standards.

### 5.3 *People*

Migration of master data work to the service centres continued and included legacy processes as well as the standardised global processes. A critical mass of 50 % of master data activity was reached in 2008. This critical mass enabled easier implementation of continuous improvement methods that rely heavily on input from the operations staff.

Service centre staff were trained in the master data processes relevant to them. The Streamline programme included extensive training to all staff from the high-level processes to the detailed use of the ERP system. Training was targeted methodically using a “role-mapping database” that mapped individuals to the roles defined for each process. Following on from the issues with MRD process delays, the MRD team performed “sanity checks” on the role database to ensure that key business roles such as requestors and approvers of MRD requests were all assigned and also that they were assigned to a reasonable number of individuals. Too many people was as much an issue as too few.

For the legacy processes, the methodology for migrating work to the service centres followed a structured approach to ensure the successful transfer of knowledge. The migration progressed in stages that included identification of the scope of work to migrate, quantification of the volume of work, handover of detailed process documentation and then progressive knowledge transfer. Knowledge transfer began with formal training sessions followed by work shadowing where the trainee observed the incumbent who explained their activities. In the next stage of parallel operations the trainee performed the activities whilst being observed by the incumbent who would correct errors and provide guidance. The incumbent would also be performing the process to maintain throughput. After this stage the trainee became the incumbent but their predecessor would continue to be available to offer support and clarify the nuances of the process.

The service centres also received training on continuous improvement methods and participated in projects to stabilise the global processes. During the stabilisation period there were suspicions that the inexperience of the service centre staff was contributing to the issues. The structured approach to analysing the issues found that in most cases the service centres were performing as required. The rigorous migration methodology instilled a culture of adherence to process that was well suited to the establishment of global standard processes. The service centres, rather than being a source of issues, had acquired a deep knowledge of the processes and would play a greater role in providing support to all players in the process.

Some of the legacy processes relied on the deep knowledge and experience of the staff in the operating unit. This was raised as an impediment to the work migrating to the service centres. Analysis of the work showed that at least 80 % of the work could be proceduralised and migrated leaving 20 % to be performed in the operating unit. Furthermore, by continuous improvement of the processes and knowledge transfer to the service centres, that could also be migrated over time.

## 6 Becoming a Process (2008–2012)

Having established consistent MRD process performance, the next step was to further improve on the MRD processes and apply those improvements across all data types. This step on the maturity model required changes to enable continuous improvement methodologies to be applied to data as an end-to-end process.

### 6.1 Processes

In 2008 Shell's Finance function began a programme to improve performance to world class. Process teams had responsibility for the design, operation and continuous improvements of the processes. Included among the processes was the data process with responsibility for managing master data.

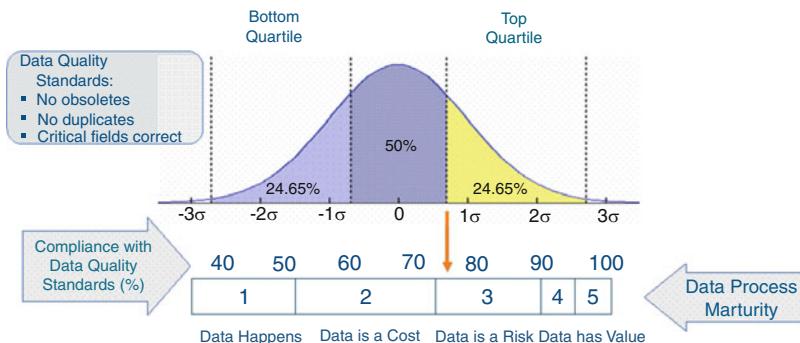
Strong emphasis was placed on continuous improvement using Lean Sigma methodology which required processes for identifying improvements and executing improvement projects. Every process team had aggressive targets set for improvements which were monitored closely to ensure the benefits of the investment were realised.

Process issues are first handled according to their urgency and are also recorded in an issues log. The issues log is reviewed regularly to identify where process improvements may be needed such as recurring issues. The improvements are prioritised based on their impact which could be a hard bottom-line saving, other quantified benefits or unquantified or “soft” benefits. The improvement projects are executed as continuous improvement (CI) projects that require a full analysis of root causes to develop a solution, solution implementation (SI) projects where a known solution is available or operational improvement (OI) projects for simple fixes that can be implemented by operations staff.

Improvements to a process often result in changes to the design documentation. The improvement process needs to include change control, document management and communications. Change control ensures that the changes are properly approved. Document management makes the documents accessible to the process performers. Communications are needed so that all stakeholders are aware of the changes.

Improvement to world-class performance for master data management requires measuring progress. Benchmarks for master data management are not established but progress can be gleaned from maturity models, surveys, consultants, conferences and networking with other companies. Unfortunately, “hard” benchmarks are rare if not nonexistent so benchmarking is still subjective.

One could infer from the Gartner Group strategic planning assumption of 2007 [2] that companies above level 2 (reactive) on their maturity model by 2010 are “top quartile” as shown in Fig. 3. A comparison of maturity assessments of parts of the organisation and the levels of measured data quality implied that



**Fig. 3** Data quality at level 2. Gartner Group Strategic Planning assumptions: “Through 2010, more than 75 % of organizations will not get beyond Levels 1 and 2 in their data quality maturity (0.8 probability)” [2]

data quality measured as around 75 % compliance to standards was an indicator of level 3 maturity and therefore “top quartile”. Indeed, the implementation of a quality measurement programme is an indicator or maturity beyond “reactive” Shell’s experience was that implementation of data quality measures initially showed around 50 % compliance to standards. Implementation of operational data management processes to apply standards and data cleanse of the database would quickly result in 75–80 % compliance.

Calculating the cost of poor quality data continued to be elusive. Improvement projects utilised Failure Mode Effect Analysis to produce objective measures of the impact of poor quality data but not in monetary terms.

Tom Redman, “The Data Doc”, says that a database is like a lake. To clean it up you first need to remove the source of pollutant [3]. Quality efforts to date had focussed heavily on the “lake” for both measurement and clean-up. An underlying assumption was that the new business and master data processes were producing clean data because validations against business rules were embedded in them and rigorous implementation ensured correct execution.

Measuring the “polluted river” utilises the existing automated “lake” measurements. Comparison of the current period results against the previous period enables identification of data that was created or updated. The comparison was only possible with the automated measurements, as it required record-by-record comparisons that are not possible with a sampled measurement.

Initial measurements showed that the data coming from the “river” is not as clean as assumed. In some cases it was found that the effort that was expended in cleaning the “lake” was fully negated by the addition of new errors from the “river”. A key reason identified was fragmented documentation of the business rules. Current efforts are now focussing on improving the preventative controls in the operational processes and improving the management of metadata.

## ***6.2 Technology***

The original implementation of the DQT required considerable effort from IT whenever the business rules were improved. The effort went into upgrading programmes to extract data from the ERP systems and to writing the measurement queries. The effort and cost required prevented the automation of QC for a number of data objects and systems so they could not benefit from frequent, complete checking of both “lake” and “river”. A new version of the DQT was developed which uses a data driven method to define the extracts and an end-user query tool enables data practitioners to develop and update the queries. The new version made it possible to extend the use of the DQT to legacy ERP systems and to a broader range of master data objects. By the end of 2012 nearly all data objects will have automated checking of their DQS across all major ERP systems.

## ***6.3 People***

The data process was led by a process executive with a team of senior managers comprising a process owner for the master data process and data managers with operational focus managing the interfaces to stakeholders in the businesses and functions.

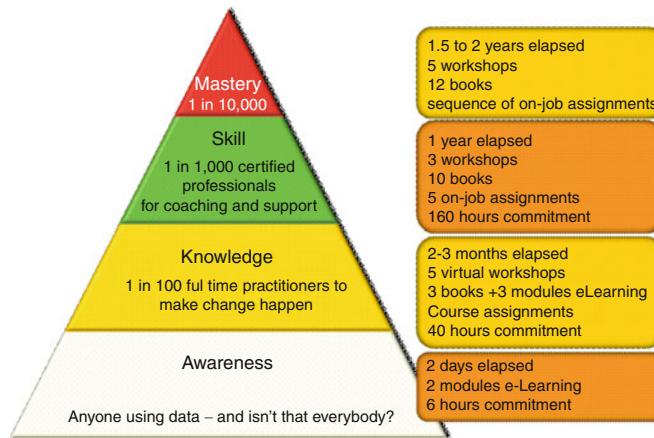
An extensive Lean Sigma training programme developed staff in all processes. Each process had targets set for yellow belts, green belts and black belts. The training and the improvement targets were designed to bring about behavioural changes to continually seek out improvements.

The data managers were responsible for identifying improvements beginning first with improvements to their operational processes then progressed to finding business benefits accruing from high-quality data once the easier operational improvements had been exhausted.

A competency framework profiles the competencies and proficiency levels a person needs to perform their job. Proficiency levels include Awareness, Knowledge, Skill, and Mastery. Individuals are assessed against the relevant competency profile to identify where they need development.

Achieving world-class master data performance requires developing and sustaining a pool of skilled practitioners. Master data management has often relied heavily on a small number of individuals with deep expertise who are stretched thinly. Their efforts need to be channelled into providing guidance to a cadre of trained specialists who understand:

- Business processes that generate data
- Business processes that use data
- Best practice data process



**Fig. 4** Competence development for data quality

In 2010, Shell began developing a training programme for data practitioners. The training programme consisted of training courses, competence assessments and a training schedule.

Training courses target the awareness and knowledge proficiency levels. Knowledge is the level at which an individual can perform the tasks for a competency. At awareness the individual has an understanding of the principles and can begin to apply them with assistance. This is a necessary step towards knowledge.

Continuous improvement is a prerequisite to much of the data quality training. Data practitioners follow normal Lean Sigma development undergoing “yellow-belt” training which teaches the fundamentals of Lean Sigma and enables them to work on continuous improvement projects. This is a prerequisite to “knowledge” training in Data Quality. They progress onto “green-belt” training which enables them to manage continuous improvement projects and to use the tools and methods.

Data quality training builds on Lean Sigma training. New starters undergo basic “awareness” training which introduces them to the principles behind their processes and enables them to perform their processes with guidance. They then progress onto “knowledge” training which connects the theoretical to practical application and enables them to perform their work largely unaided.

A successful data quality programme needs a critical mass of people at different levels of proficiency as shown in Fig. 4. Being sustainable requires a further understanding of the rate at which people progress through the ranks. A once-off training programme is not enough. A development programme needs to target a percentage of people at each level every year to maintain the critical mass.

Awareness training consists of an in-house developed 1-h e-learning course that is relevant to anyone working with data and a further externally supplied 5-h e-learning course for data specialists. The in-house course explains why data quality is important and introduces the idea that everyone has a role to play in improving

data quality. The externally supplied course covers a broader range of topics across the span of Information Management.

Knowledge training is a blended learning course comprising three e-learning modules, reading from three books, five virtual workshops and five graded assignments for a total of about 45 h time commitment over 3–6 months. The training uses a diversity of sources so that participants are exposed to a range of perspectives on data quality.

The staff need to be assessed regularly to identify any gaps from their target proficiency level or their readiness for further development. This helps with prioritising training delivery. Awareness is checked with a multiple-choice questionnaire of material from the awareness courses. For knowledge, individuals self-assess using a questionnaire based on the assignments from the knowledge course. The questions are of the form “are you able to...”. Their responses are validated by their line manager. For all assessments, the line manager confirms the proficiency level using the assessment questionnaires as a guide.

The training schedule is designed to maintain a target proficiency profile across all master data practitioners. A once-off training programme is not sufficient as service centre staff progress through the ranks, move to other roles within the company or leave the company. The scale of the service centres requires an ongoing training programme to be in place.

Data practitioners need to be expert in the business processes and the data objects they work with. A structured approach was implemented in 2012 to develop proficiency in:

- The data processes they work with
- The data objects they work with
- The business processes they interact with
- The businesses and functions they interact with

For data customers we have developed training material to raise awareness of data quality and to explain the role that individuals play in ensuring data quality. In particular, describing their requirements and providing feedback on data quality and its impact on their work. Awareness training consists of an in-house developed 1-h e-learning course that is relevant to anyone working with data.

We are developing a training programme for people working at the interface between business processes and MRD processes. This includes business process owners and process managers who provide the requirements that define data quality. The training teaches them the activities they need to perform to get continuous improvement of data quality.

## 7 Beyond 2012

The next challenge is to seek even greater improvements in data quality in support of business performance. This means adopting more advanced methods for managing quality and further development of proficiency in our staff.

## 7.1 Processes

Calculating the cost of poor quality continues to be difficult and is critical to making further improvements to master data processes. Increased involvement of data practitioners on business improvement projects should help to make the connection between business issues and poor-quality data. Work is underway to define in more depth the management roles in improving master data processes and data quality with increasing emphasis on defining and exploiting the value of high-quality data.

Poor quality metadata has been identified as a root cause of many data quality issues and needs to be managed. Metadata is often fragmented and developed primarily to meet the needs of IT projects. Metadata needs to be managed like any other data to make it useful for data practitioners in their daily work. This means that the end-to-end processes for managing metadata need to be defined and implies that roles such as process owner and process manager for metadata are needed. Measures of metadata quality and metadata process performance are essential elements of the metadata process.

Better objective measures of performance, benchmarked against best practice, are needed to drive ongoing improvements in the details of the master data processes. This will require collaboration across companies, facilitated by an intermediary like a professional body, to set standards, collect data and produce benchmark reports. To get to this stage, a large number of companies would need to achieve levels of maturity where they are regularly measuring data quality and process performance across a range of data objects. Current indications are that few companies are at this stage so this level of benchmarking is still some time off.

## 7.2 Technology

MDM solutions are maturing and will help support improvements through automation of master data processes. Data objects and content that spans the main ERP systems are candidates for an MDM solution. But in the absence of any urgent need for an MDM solution, it remains only to monitor the marketplace to assess the maturity of the offerings.

Metadata management needs tools that support the wide range of metadata needed to manage master data quality. In line with the approach to managing master data, the first priority is to develop simplified and standardised processes for managing metadata. Automation of metadata management has, at face value, similar requirements to master data management. Tools for managing metadata as an end-to-end process would appear to be even further away than MDM.

### 7.3 *People*

The service centres will continue to build proficiency with training and experience. Having been in existence for some 8 years now, some of the original recruits have vast hands-on experience in managing data quality and are moving into senior roles. The next stage of proficiency is “skill” where a person has a deep understanding of the principles and can apply them successfully in a range of situations. They can also provide direction and coaching to others.

About 10 % of staff need to be developed to skill level. Skill-level training needs to cover more advanced methods for improving data processes and data quality to enable higher work performance and coaching and guidance of other staff. Training on data quality methods needs to build on Lean Sigma training to green-belt or black-belt level.

Professional certification is critical to ensuring ongoing development of data practitioners. People at skill level provide and coaching to the less experienced and it is vital that they are communicating best practices to them. Professional certification such as IQCP (Information Quality Certified Professional) as offered by the IAIDQ (International Association for Information and Data Quality) ensures that keep up to date with current best practices.

Service centres also need to be able to deliver training to large numbers of staff. Knowledge level courses require facilitation by data practitioners, preferably at skill level or higher, who are able to deliver the training effectively. They need to be able to present training material, facilitate workshop discussions and assess coursework. A “train the trainer” programme has started to develop people who can deliver the knowledge level training.

Business people who interface with the MRD processes need higher proficiency in data quality management. They need, at minimum, training to knowledge level that is targeted at their particular role in data quality.

## 8 Summary of Key Points

The key element of Shell’s data quality programme has been to consolidate the data quality expertise by creating full-time roles and migrating them to a central organisation with a strong focus on data quality.

Greatest attention has been placed on operational activities. Benefits of data quality come from business processes and are the direct result of the operation of data processes. These processes need to be well designed to be efficient and effective and must be well documented to ensure consistent execution. Establishing process performance measurements and continuous improvement of business processes and data processes ensures ongoing delivery of benefits from data quality.

A data quality programme is greatly facilitated by large business re-engineering programmes. A process centred approach to data quality based on measurement and

continuous improvement, apart from being the most effective way to make progress on data quality, also helps with alignment to the business programme. A notable feature of master data management in Shell is the wide range of data objects that are covered. This was made possible by the breadth of the Streamline programme which re-engineered all the Downstream business processes.

When processes have been simplified and standardised, automation with tools helps to make them more efficient. Work flow management, data quality reporting and bulk updates tools play an important role in making master data management more efficient.

Change management at all levels of the organisation is a critical element of a data quality programme. Key lessons to aid this are:

- Identify and deliver business benefits through the data quality programme through bottom-line savings and better utilisation of business resources.
- Be mindful of business concerns over management controls at the coalface.
- Address the risks and demonstrate how you mitigate them while still delivering benefits to gain ground level support for the data quality programme.

Finally, develop the competences in data quality with formal training. Aim to develop data management staff into experts in their own right by addressing not only the technical aspects of their processes but also principles of data quality management. Business people also need training on the principles so that they too can be effective in driving data quality improvements.

## References

1. Arnheiter ED, Maleyeff J (2005) The integration of lean management and Six Sigma. Emerald 17
2. Bitterer A (2007) Gartner's Data Quality Maturity Model. ID Number G00139742. <http://www.gartner.com/technology/home.jsp>
3. Redman TC (2008) Data driven: profiting from your most important business asset. Harvard Business Press, Boston

# Creating an Information-Centric Organisation Culture at SBI General Insurance

Ram Kumar and Robert Logie

**Abstract** For an insurance business, data is its lifeblood. It drives most, if not all, significant decisions including product design, pricing and marketing. Without ‘good’ data, an insurance business is almost blind. No matter how smart and efficient business’s processes are; how advanced, savvy and solid the IT systems that support the processes are; and how capable and skilful the staff who use the processes and technology are, if the underlying data and information that these processes, technology and people use is not good enough in terms of its quality and integrity, the outcome such as effective and efficient decision-making will be poor. The strategy of a business should recognise that information assets, supporting technology, business processes and people need to be coordinated and managed effectively. This chapter is about an award-winning case study of a general insurance business but applies equally to most businesses, regardless of industry. It tells the story of how the organisation created an information-centric culture by bringing four objectives relating to technology, business process, people and information to work together in a collaborative manner.

## 1 Introduction

Imagine what it’d be like if every decision was based upon quality, up-to-date information? Where everyone trusts the data they use? What would it be like if everyone using the data consistently understands the meaning of that data? Where decisions are taken faster than ever before? And if the relevant information you

---

R. Kumar (✉)

Insurance Australia Group (IAG), Level 19, 388 George Street, Sydney, NSW 2000, Australia  
e-mail: [ram.kumar@iag.com.au](mailto:ram.kumar@iag.com.au)

R. Logie

SBI General Insurance Company Limited, Mumbai, India  
e-mail: [rob.logie@sbigeneral.in](mailto:rob.logie@sbigeneral.in)

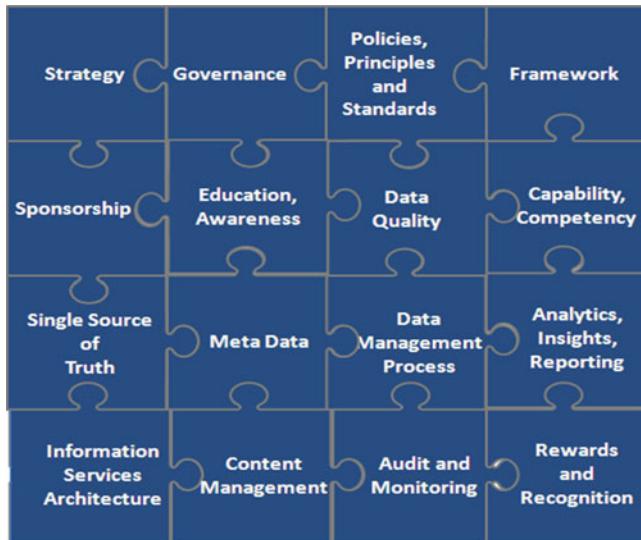
needed is easily available in a timely manner and need not be hunted down? You're imagining a world that has recognised the value of information management. It's no less true for being repeated, but for organisations their information and data are their greatest assets. Unfortunately these assets often underutilised because many organisations don't recognise information and data as an organisation asset or entity in itself, and therefore, they either take information and data for granted or they don't know how to manage them effectively.

Organisations are challenged with many serious information management problems that are increasing the cost of its operations in many ways and at the same time reducing its productivity, efficiency in its operations, growth and challenges in meeting and adapting to business and customer demands and growth. The last decade has witnessed significant growth in electronic information, and this had only made the information management issues and problems of organisations worse. The future is a digital age as the growth volume of data is exploding due to reasons such as social networking and various sensor data, and this is putting significant pressure on the organisations to tap into this external data world for better analytics and decision making, on top of the increasing requirement to manage its internal data.

Organisation information assets are contained in the form of data and records and in the knowledge, experience and judgment of its people and intermediaries. Along with people, an organisation should view information as a key asset. As part of an organisation's strategic objectives, it should recognise that information assets, supporting technology and business processes and people need to be coordinated and managed effectively. Underpinning this approach are the legal requirements set by government to ensure organisations manage their resources effectively, ethically and efficiently.

There are still different views of what enterprise information management is. Too frequently the term is used for any project that is related to managing information and therefore associated with a specific project or application such as a data warehouse/business intelligence or content management system. In these projects, the issues that a 'siloed' approach can have on the consistency, transparency or reusability of information across enterprises are often ignored. Also, with this approach, enterprises ignore that they fail to achieve efficiencies among similar information management efforts that exist inside the enterprise. In other terms, although organisations have been performing information management activities (data quality, metadata management, data integration, content management and so on) for quite some time, we find that most of these efforts are done for a specific functional area or department. While such information management practices help meet local needs, they foster information silos—the inability to share and exchange information across the enterprise. What distinguishes information management from enterprise information management is the ability to leverage information across teams, groups or even with business partners to spur collaboration, innovation, transparency and agility.

An information/data management strategy comprises of many components that need to be tackled together to get the most effective, efficient and sustainable value out of it. It has several moving parts like components of a jigsaw puzzle that



**Fig. 1** Information management puzzle

should be addressed individually and connected together to achieve the sustainable value. Figure 1 shows the different key components of an information management strategy.

Figure 2 outlines some of the well-known and common problems an organisation may face due to lack of an enterprise wide strategy for information management. On the flip side, some of the key business benefits in defining and implementing an information management culture and strategy are:

- Significant long-term cost savings for the organisations due to efficient and effective use of the information assets.
- Ability to understand and identify growth opportunities by using reliable and quality data and information.
- Greater efficiency in implementing and managing business and information processes.
- More valuable, superior, high-quality, timely and powerful information for efficient and effective decision-making.
- Market leadership in information management through creation of a high value asset.
- Quicker delivery of business products to markets through supporting Information Technology.
- Better risk selection and management through well-managed data improves policyholder risk profiling and fraud prevention and aligns financial risk decisions with overall corporate strategy. Accurate, timely and harmonised data provides trust in information for underwriting to run risk/pricing models as well as finance to project and reconcile enterprise financials with confidence.

- Better regulatory legal compliance due to the confidence on the data used by the organisation.
- Ability to integrate and utilise the external data assets effectively through the streamlined internal processes for data management.
- Increased productivity of staff by using timely and reliable information.
- Reduce technology integration costs as data is shared and reused consistently.

<b>Information/Data Silos</b>
<ul style="list-style-type: none"> <li>• Large number of disparate information management systems (developed over a period of time as business grew or through mergers and acquisitions) addressing specific business problems thereby producing silos of information resulting in poor quality of data (e.g. duplicated data, data integrity)</li> <li>• Little or no reuse of information assets collected due to lack of analysis of the value the collected data provides to the organisation</li> </ul>
<b>Business</b>
<ul style="list-style-type: none"> <li>• High cost of maintaining information/data in different places in an inconsistent and duplicate manner</li> <li>• Poor decisions due to poor quality information/data</li> <li>• Lack of availability of timely and reliable data</li> <li>• Poor customer and other stakeholder experiences</li> <li>• Lack of understanding of how the business is performing</li> <li>• Unable to make accurate business decisions</li> <li>• Inability to accurately measure business units and individual's performance – also different measures for the same metric</li> <li>• Inefficiency of processes due to poorly supported data</li> </ul>
<b>Integration Challenges</b>
<ul style="list-style-type: none"> <li>• Serious integration problems between information management systems resulting in expensive integration project costs</li> <li>• Information system specific data and information models that is hard to integrate.</li> <li>• Lack of enterprise wide data model with supporting standard data dictionary to enable interoperability, sharing and common understanding of information across the organisation</li> <li>• No reuse of data assets and lack of master data /single source of truth for the enterprise resulting in expensive integration costs leading to solutions that are point to point</li> </ul>
<b>Data Quality and Integrity</b>
<ul style="list-style-type: none"> <li>• Poor quality of data and information, including lack of consistency, duplication, and out-of-date information and data</li> <li>• No common business definition and models for data management</li> <li>• No metrics to measure the quality and value of information assets</li> <li>• No ownership and classification of data</li> </ul>

**Fig. 2** *continued*

<b>Technology</b>
<ul style="list-style-type: none"> <li>• Multiple information systems addressing same business requirements across the organisation due to lack of coordinated effort or understanding of existing systems resulting in pockets of duplicate data</li> <li>• Increasing legacy of information systems that require replacement or upgrade resulting in expensive maintenance costs</li> <li>• No strategy to manage the use of technology to support information needs of the business. Technologies are introduced as required to address a specific business requirement</li> <li>• Information assets locked into vendor dominated proprietary technologies that result in solutions that are not scalable, adaptable and flexible to meet business change requirements</li> <li>• Limited resources for deploying, managing or improving information systems</li> </ul>
<b>Organisation and People</b>
<ul style="list-style-type: none"> <li>• Little or no recognition and support for information management (end to end information lifecycle) by senior executives as they fail to understand that information is a corporate asset</li> <li>• Internal politics impacting on the ability to coordinate activities across the organisation</li> <li>• Lack of understanding of information management value by staff</li> <li>• Limited and patchy adoption and use of existing information systems by staff</li> <li>• Lack of knowledge management strategy which includes people, process, information and data</li> <li>• No common understanding of business terms across the organisation due to lack of enterprise wide thesaurus/data dictionary</li> <li>• Large number of diverse business needs and issues to be addressed</li> <li>• Lack of clarity around broader organisational strategies and directions</li> <li>• Difficulties in changing working practices and processes of staff</li> </ul>

**Fig. 2** Implications of lack of an enterprise information management culture and strategy

- Improve consistent use of information, information reuse, sharing and interoperability.
- Improve work practices through better access to quality data in a consistent and standard manner.
- Improve organisation's responsiveness to customer, stakeholders and partners.
- Enhanced customer support and increased customer loyalty. Focusing support and retention efforts driven by customer insights can easily increase profitability per customer, rather than acquiring new customers.
- Financial transparency and performance management. Cut through the complexity gain financial transparency at all levels by enabling products, claim and underwriting performance analytics through clean, timely, integrated data. Data lineage and harmonisation for consistent enterprise performance reporting yields such as improvements in loss-costs, and pricing and capacity optimisation.

- Increased profitability and customer's share of wallet penetration by creating a new customer experience model that includes customer insights and life events. Through the management of this information you can achieve transparency of customer's life events, household, transactions, product holdings and legal hierarchies and measure account performance accurately.
- Improved sales and marketing effectiveness by creating a unified and accurate customer view that drives business actions that relate specifically to customer segments including offers, life event marketing, discounts and availability of information for the customer.

## 2 Creating an Information-Centric Organisation Culture

This chapter discusses a case study about strategising and creating an information-centric culture in a general insurance organisation in India. SBI General Insurance Company Limited (<http://www.sbigeneral.in>), India, is a joint venture organisation jointly created by the State Bank of India (<http://www.sbi.co.in>), India's largest and biggest bank, and the Insurance Australia Group (<http://www.iag.com.au>), the no.1 insurer in Australia and New Zealand.

SBI General's current geographical coverage extends to 37 cities pan India. It is currently serving 3 key customer segments, i.e. retail segment (catering to individual and families), corporate segment (catering mid to large size companies) and SME segment. Current Policy offering of SBI General covers Motor and Home Insurance for Individuals and Fire, Marine, Package, Construction and Engineering and Group Health and Miscellaneous Insurance for Businesses. It commenced business (in a limited manner) in July 2010 and the IT system went live in March 2011. The organisation has so far generated revenues in excess of USD130 million and has a staff of about 1,200+ and is growing at a significant pace.

State Bank of India (SBI) is the largest banking and financial services company in India by revenue and total assets. It's a state-owned corporation with its headquarters in Mumbai, Maharashtra. The bank traces its ancestry to British India, through the Imperial Bank of India, to the founding in 1806 of the Bank of Calcutta, making it the oldest commercial bank in the Indian subcontinent. Bank of Madras merged into the other two presidency banks, Bank of Calcutta and Bank of Bombay to form Imperial Bank of India, which in turn became State Bank of India. The government of India nationalised the Imperial Bank of India in 1955, with the Reserve Bank of India taking a 60% stake, and renamed it the State Bank of India. In 2008, the government took over the stake held by the Reserve Bank of India.

SBI provides a range of banking products through its vast network of branches in India and overseas, including products aimed at non-resident Indians (NRIs). The State Bank Group, with over 25,000 branches, has the largest banking branch network in India and over 200 million customers and 50,000+ customer touch points (e.g. ATMs). It also has around 130 branches overseas.

With an asset base of \$352 billion and \$285 billion in deposits, SBI is a regional banking behemoth and is one of the largest financial institutions in the world. It has a market share among Indian commercial banks of about 20 % in deposits and loans. It is growing at 25 % every year. The State Bank of India is the 29th most reputed company in the world according to Forbes. Also SBI is the only bank featured in the coveted ‘top 10 brands of India’ list in an annual survey conducted by Brand Finance and The Economic Times in 2010.

Insurance Australia Group Limited (IAG) is an international General Insurance Group with operations in Australia, New Zealand, Asia and the United Kingdom. IAG businesses include some of the world’s most trusted and respected general insurance brands. IAG is the biggest and leading General Insurance Group in Australia and New Zealand.

IAG underwrites around \$9 billion of insurance premiums each year. The group insures over \$1,000 billion worth of property and employs around 14,000 people. IAG Group sells insurance under many leading brands including NRMA Insurance, RACV, Swann, CGU, SGIO, SGIC and The Buzz (Australia); NZI, State and AMI (NZ); Equity Red Star, Equity Broking and Barnett & Barnett (UK); NZI and Safety (Thailand); SBI General Insurance (India); AmAssurance and Kurnia Insurance (Malaysia); AAA Assurance (Vietnam); and Bohai Insurance (China). IAG is amongst top 40 listed companies on the Australian Stock Exchange in Sydney.

Data and information are the lifeblood of insurance industry. The insurance markets in which we operate are being commoditised. We can only differentiate ourselves on price and/or service. Both price and service are underpinned by business process. Efficient business processes help to reduce costs and enable us to offer competitive and profitable products. Customer-oriented processes make it easier for us to attract and retain customers. In order to achieve competitive advantage, we will in general not share any of our key intellectual property around pricing or process. However, we should endeavour to share our internal intellectual property, learn from each other and leverage our efforts. The efficiency of our business processes and our supporting information management systems are reliant on how we effectively manage the information life cycle of our organisation in terms of planning, collection, storage, access, use, archiving and destruction.

Insurance industry is in general facing tough challenges in increased competition from banks and brokers, shrinking price premiums, added complexity and customer demand in product offerings, stricter government regulation, sluggish economy and ageing workforce. Consider these trends in the insurance industry [1], as identified by EMC’s financial services research, Insurance Networking News, AAXIS Group Research and Forrester:

- 18 % of total claims payments are fraud.
- The industry tops US \$600billion in data quality costs.
- 30 % of insurance IT time is spent fixing data.
- Insurance underwriting leakage grows to 9.7 %.
- The underwriting workforce is ageing.

- Many insurers observe the inefficiency of direct marketing programs and low conversion rate.
- Regulatory and legal requirements are becoming stricter (e.g. new medical claim reporting requirements and Solvency II reporting requirements in Europe).

In addition, consider the new market and competitive forces [1]:

- Strong focus on risk-based pricing.
- Actuaries play a key role in an insurance business to stay competitive.
- Events monitoring and predictive analysis for product, risk and pricing decisions.
- Online commerce (including self-service) and social media increase impact on the marketing and sales of policies.
- Dramatic growth in data inputs from sensors and other instruments enables monitoring of assets under insurance to provide enhanced risk patterns.
- Pressure has increased to grow market share without sacrificing profitability regained from premium rate increases during the recent market conditions.

Let's look at some of the typical data issues in insurance industry [1]:

- Information for customers may be different in the marketing, underwriting, claims and finance applications. This causes customer satisfaction issues and opportunity losses.
- With customer and household information spread across several systems, either as a result of legacy systems or systems inherited through mergers and acquisitions, it's very challenging for insurance companies to gain a complete view of customers and the relationships to other individuals.
- Consistent product definition can be particularly vexing. Ideally products on the underwriting side and claim side should be identical, at least at a high level, allowing product-level loss run analysis.
- Similarly, an insurer may use brokers for direct policy brokering, reinsurance brokering and claim brokering. Large brokers will be used in all scenarios and may be defined repeatedly and in different ways in each system.
- Disparate chart of accounts and financial codes across lines of business make it very difficult to roll up or match balances and transactions for enterprise reporting.
- Significant data quality issues and conflicting data and semantics (metadata) exist within and across data sources.

Information/data flows across the organisation and not in silos. The insurance value chain of the organisation does not work in silos or vertically but horizontally across the organisation. Therefore, it is important for the organisation to think horizontally rather than vertically to create efficiency and effectiveness in terms of processes, data, information technology and others.

This is where the culture of end-to-end thinking when it comes to data and processes is critical in an organisation. This point is often missed by organisations and, in particular, between business and information technology. The role of information technology is to support and provide sustainable value add to the other

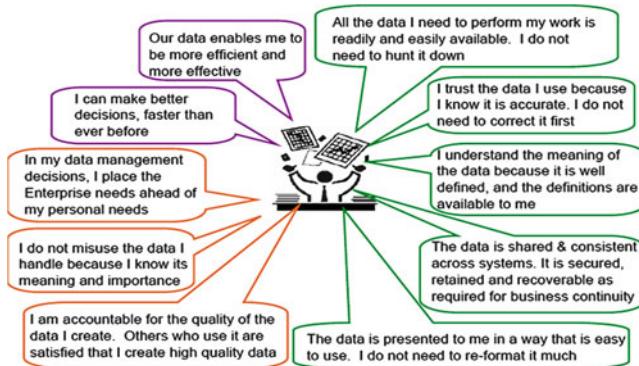
parts of the organisation. Information technology is now becoming a commodity and cannot be seen as a cost anymore. It is as important as any business departments such as sales and marketing, human resources and accounting. Information/data is the ‘common denominator’ or the ‘bridge’ between the business and IT as the input to the business is data and information and the output from the business is data and information.

Perhaps the most interesting aspect of this case study is how we decided to undertake such a comprehensive approach to enterprise information management. There were a number of converging and serendipitous drivers for our approach. First and foremost was that we were starting a new business—a real green field—no people, no processes, no income, no costs and no legacies. Secondly, we had a person on the team who not only was passionate about information management but also had the requisite skills and experience to take theory and put it into practice. Thirdly, we had a senior management team that wanted to build a sustainable business, one that would stand the test of time. Lastly, we had the support of shareholders who saw the long-term value in the approach.

Lucky you may say, perhaps. But looking back, we think we will implement the same components in any business we might work with in the future. Once a business has experienced the power of a holistic approach to information management you really wonder why every business does not make the effort to do it. Often organisations with legacy systems find excuses not to implement an information management strategy by citing complexity with the legacy systems and the data issues. But what organisations need to realise is that information management strategy and execution is a journey and cannot be implemented big bang and requires commitment and determination at the highest level in the organisation if the organisation values its information assets. For example, any organisation with legacy often implement large transformation program whether it is business- or IT-related transformation. Defining a comprehensive information management strategy for the program and execution of the same are great start to the information management journey. But organisations often ignore this citing budget and time as constraints that would often result in the same or even more complex information management problems in the future.

The case study holds nothing back as far as what SBI General has done. As with most business processes it is the implementation approach that matters. Here you will find a description of the key information management components and their purpose, but implementation remains the secret to success. Our ultimate objective is to have a business that doesn’t even realise that it is getting its information management right; it is just the way we do business.

Hopefully, this case study will inspire some other companies to take the challenge, piece by piece, and reap the benefits of the wonderful world of effective enterprise information management. The following sections explain what key strategic programs were initiated and implemented to commence the journey of creating a sustainable information management culture at SBI General.



**Fig. 3** Outcome of an information management culture

### 3 Strategy and Governance

A solid information management strategy and execution of the same coupled with strong governance with support and sponsorship from the top are critical to establishing a sustainable information management culture in an organisation. If senior management of an organisation do not understand the value information management brings to the organisation and ensure the information assets are managed effectively and efficiently, nobody will and even if they do, execution of information management strategy could still be a constant challenge in the organisation unless an information management culture is inculcated into the organisation. Information management culture development process involves defining a clear, concise and simple goal for information management in the organisation. The information management goal of the organisation is defined as

Get the Right data to the Right place at the Right time in the Right format with the Right quality in the Right context with the Right security and, with the Right governance [2].

How do we measure success in terms of creating an information management culture in the organisation that brings ability for people to be productive in their work and to make sound, effective and efficient business decisions? Figure 3 demonstrates the outcome that the organisation wants to create and be measured on [3].

The journey to an information-centric culture requires clear planning and execution that does not happen overnight. Following are the activities that were undertaken at SBI General to make this happen.

#### 3.1 Education and Awareness

Educating and raising the awareness of information management to all employees, right from senior executives to data entry operators, is critical. A 2-h information

management awareness pack was created that covered basic definitions of data and information management to defining the entire information lifecycle management. This education and awareness training requires commitment from senior executives by leading from the front. This training should be done both top-down and bottom-up in an organisation to create an information management foundation and understanding among its employees.

All employees including branches of the organisation sat in this training. This has now resulted in the training program now being integrated into the human resources induction program that is conducted for any new employees joining the organisation.

In addition to the information management awareness program that is now an integral part of the organisation's employee induction program which is compulsory for new employees to attend, in order to spread greater awareness and need for data quality, SBI General has introduced the concept of a mascot which symbolises and promotes the culture of 'Getting Quality Data First Time, Every Time'. The mascot will help the organisation at all times to remind its staff that quality of data is of utmost importance and that everyone has a responsibility towards achieving it.

### **3.2 *Information Management Strategy***

The Information Management Strategic Plan for SBI General Insurance Company Limited (SBI General) has been designed to define a pragmatic set of strategies for implementation. The approach taken was to implement the strategic plan in stages (three to six monthly cycles with a concrete set of milestones and deliverables) as opposed to a 'big-bang implementation approach'. Certain components of the strategy that meet the immediate requirements of the business, that are tangible and that could measure success were implemented first.

The following key strategic objectives of the Information Asset Management Strategic Plan for SBI General were identified. The seven key objectives identified are:

- **Information Management Framework:** To define an information management framework for SBI General. This will enable all information- and data-related activities across SBI General to be mapped within the framework. This framework will include defining the information life cycle of SBI General (more details in Sect. 3.3).
- **Information Management and Governance:** To value information as one of SBI General's core assets and take responsibility for information management and planning. This includes business units implement/follow the corporate strategy on information asset management and take responsibility and ownership of its information assets and manage them through its supporting information technology (business) systems.

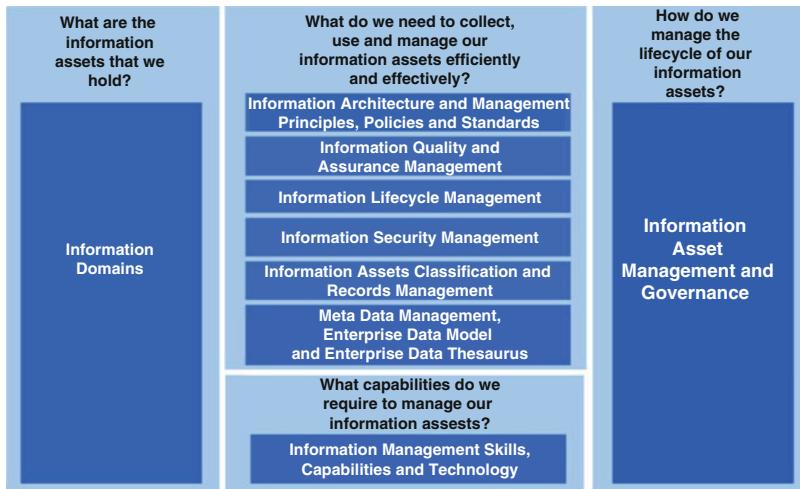
- **Managing the Information:** To provide timely, relevant, reliable, quality and consolidated information (no redundancy and duplication) in a standard and consistent manner to SBI General staff, stakeholders, partners and customers.
- **Managing the Metadata:** To ensure that the context and requirements for SBI General information are defined and used to guide the development of new information systems and integration of existing systems that support business needs.
- **Information Quality Management:** To ensure that data and information captured, processed and stored in business information technology systems meets agreed data/information quality standards, principles and policies.
- **Marketing and Communication:** To improve the understanding and use of information assets and its management by all sections of SBI General.
- **Staff Training and Development:** To ensure that SBI General staff (all business departments and IT department) understand the value of information as a core asset of the organisation and, importantly, have the necessary skills for the effective utilisation of information assets and its management strategies in their area of responsibility.

Expected outcomes and benefits for each key strategic objective were identified in the information management strategy from successful implementation in stages over a period of time. Implementing the key strategic objectives in full to produce the planned outcomes and benefits is a 2-year strategy that was implemented in stages, keeping in mind immediate business requirements and priorities and medium- to long-term priorities. The cycle of implementation was broken into 3 to 6 months.

Without a senior executive taking accountability and ownership for creating information management culture within the organisation by driving the planning and execution of information management strategy is like a plane in the sky without a pilot. Information is the core asset of the organisation and it requires executive sponsorship, accountability and support to drive the right information management and culture in an organisation and this is well understood at SBI General. Culture of information management cannot be driven bottom-up in an organisation. It requires both top-down and bottom-up to come together.

SBI General's Deputy Chief Executive Officer ('DCEO') is the '*Information Management Champion*' for the organisation and has the overall accountability for managing the information assets of SBI General. From 2013, the Chief Executive Officer of SBI General will be assuming this role. This person is the sponsor for the implementation and delivery of this information management strategy and will be supported by the business and IT departments of the organisation.

This person delegates responsibility for various elements of the strategy for execution to specific personnel. This may include recruitment of new information asset management resources or assigning the tasks to the 'Information Management Custodian (IMC)' that every business division, namely, human resources, finance, underwriting, claims, distribution, IT, sales and marketing, investments and actuarial, has. These data IMCs are accountable for managing the information/data assets of the business division, leading the implementation of information management strategy in their business division and across the organisation.



**Fig. 4** SBI General Information Asset Management Framework

### 3.3 *Information Management Framework*

The SBI General Information Asset Management Framework (IMF) (see Fig. 4) is used as the model for mapping, developing and extending any information-/data management-related work in the organisation. The key strategic objective for IMF and the planned outcomes were outlined in the previous section. The IMF serves as a key tool for data and information assets management across the organisation or within a business unit. The IMF has been developed to help rationalise the management of the organisation's/business units' information assets with the view of getting endorsed as the standard approach to information management for the organisation.

The IMF focuses on delivering useable content (business-driven data modelling, information integration, information interoperability, business intelligence, information compliance, data quality and information lifecycle management) to ensure a comprehensive and consistent approach to the management of the organisation's information resources, consistent with recognised standards and international best practice.

The framework is classified into four major categories that answer the key questions highlighted in Fig. 4:

- What are the key information domains of SBI General?
- What do we need to collect, use and manage SBI General's information assets efficiently and effectively?
- How does SBI General manage the life cycle of its information assets?
- What capabilities are required to manage the information assets of SBI General?

A set of guiding principles, policies and standards for information management were developed and were endorsed by the senior executives of the organisation

## Information and Data are Corporate Assets

### Principle

SBI General will value, manage, protect and leverage its information and data as a strategic corporate asset to improve decision making and operational effectiveness with the support of relevant technologies

### Rationale

Data and Information are valuable corporate resources; They have real, measurable business value. In simple terms, the purpose of data and information are to aid decision making. Accurate, timely data and information are critical to accurate and timely business decisions. Most corporate assets are carefully managed, and data and information must not be an exception. The role of technology is to support the use of data and information effectively and efficiently in order to meet business requirements.

### Implications

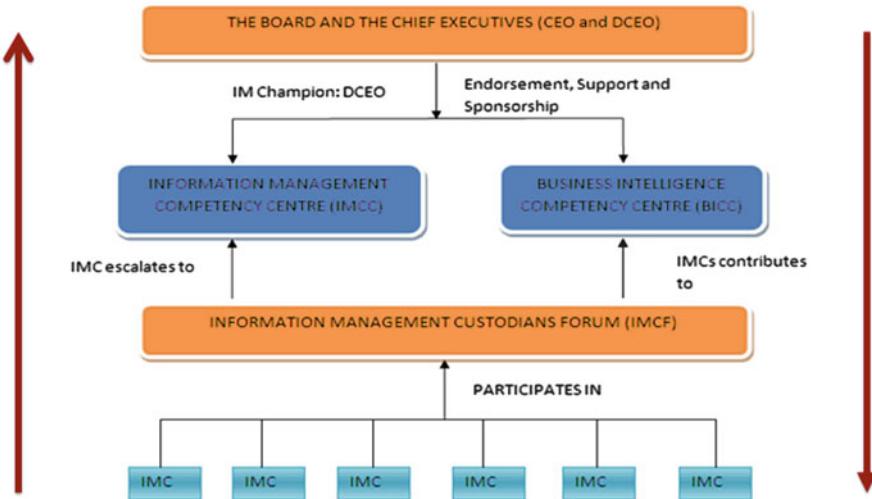
- ▶ As an asset, data has a data owner, accountable for its accuracy, and a data custodian who is responsible for its maintenance. Data owners and custodians must be identified and charged with their respective responsibilities.
- ▶ As an asset, the utilisation of the data asset must be increased to lift its inherent value. Existing and potential users of each data subject area should be identified.
- ▶ As an asset, data loses value (depreciates) over time until its value is less than the cost of maintaining it. Data that has lost value should be disposed of or archived in a way that meets legislative requirements.
- ▶ Information and data requirements to meet business requirements should be given high priority when introducing new technologies and not the other way round (i.e. giving high priority to technology over information and data).

**Fig. 5** An enterprise information management principle

including the Board. These are adopted by the organisation to bring the required discipline in the way data and information are managed and used in the organisation. Information/data management standards (5 of them) that were approved by all the departments are following/implementing these as part of corporate IM governance requirements. This includes ‘minimum standards’ for information management that all departments must comply with and that is fundamental for implementing information asset management strategy. They are business owner for data, classification of data according to significance to the organisation and retention, destruction and archiving of the data.

Each enterprise information management principle has a rationale and implications defined. These principles form part of enterprise architecture principles that cover the areas of business, information, infrastructure, application, security, integration and governance. An example of a principle is shown in Fig. 5.

Seven general principles for information management, 6 principles for operational aspects of information management, 6 principles for enterprise data warehousing, 3 principles for information quality management and 5 principles for business intelligence were developed. An information management policy along with an information security policy were developed and endorsed.



**Fig. 6** Information management governance structure

### 3.4 *Information Management Governance Framework*

Information management governance is critical for any information management strategy, and it sets up an environment for decision-making and quality improvement processes that provide the discipline, direction and support for the effective collection, use and management of information assets across the organisation. Without an effective and efficient governance framework, implementation of an information management strategy is a challenge often resulting in a tactical solution without any medium- to long-term business benefits. Information management framework, information lifecycle management process and information asset policies, principles and standards have been developed to ensure that the corporate asset is created, used, managed, monitored, audited and protected wisely. All departments are directed and measured to ensure that they implement the policies, principles and standards as everyone in the organisation has a key role to play in taking care of the corporate asset. The framework also provides the means for conflict resolution and ensures that the data that supports departmental processes is also adequate for other cross-functional or reporting needs. Figure 6 shows the information management governance structure.

Four key bodies within SBI General are charged with providing the discipline, direction and support necessary for effective management of information assets across the organisation. These are part of the governance framework and are:

- **Information Management Competency Centre (IMCC):** The key enterprise-wide body consisting of senior executives from all departments responsible for information management strategy of the organisation. Individual needs of the

business units are also addressed through this forum keeping in mind the strategic objectives of the organisation. This group is chaired by the Deputy CEO of SBI General, and members are heads of departments of distribution channel and Marketing, product and underwriting, claims, actuarial, risk assurance and compliance, finance, information technology, reinsurance, human resources, Strategy, information and performance management, and operations.

- **Information Management Group (IMG):** This group has the responsibility of implementing the Information Management Strategic Plan; developing and managing information management policies, processes and procedures; overseeing the role of the information management custodians of the business units and the information management team; assessing data quality and reporting; and coordinating the activities of the IMCC and Business Intelligence Competency Centre (BICC) thereon. The group is part of the Strategy, Information and Performance (SIP) department. Often in organisations, an information management group sits under IT department, and if it is outside of IT, it sits under finance or other groups. SBI General's view is that any new business strategies have implications on data and information requirements and hence, requires holistic and end-to-end view and impact to better manage and execute the changes. Therefore, the IMG sits under the SIP department. The head of the department manages IMG and reports to the CEO of the organisation.
- **Business Intelligence Competency Centre (BICC):** The key enterprise-wide body responsible for business intelligence across the organisation and consists of representatives from all departments listed under IMCC. This group is chaired by the head of Strategy, Information and Performance department. Business intelligence (BI) implementation involves having the means to address the organisational dynamics associated with competing BI priorities, using the technologies skilfully to gain the insight needed to make better decisions and integrating diverse applications, technologies and information.

Due to the lack of a cohesive strategy, many organisations have created multiple uncoordinated tactical BI, performance management and analytics implementations, resulting in silos of technology, skills and processes. Many organisations have also unwittingly created the same disconnected silos of people because of different expectations and requirements regarding the role of business intelligence. These organisational barriers limit the achievement of business goals by business and management and leave IT people and architects wary of supporting the business intelligence effort. It is important to consider and understand these factors in relation to why disconnections happen and the negative impact they can have on realising the value and benefit of business intelligence. These factors can also increase the TCO of business intelligence for the whole organisation.

To address unmet business needs, many enterprises now realise that they must have a strategic initiative, one driven jointly by the business and the IT organisation. A Business Intelligence Competency Centre (BICC) for an organisation is a strategic initiative that addresses these challenges.

The BICC of SBI General aims at focusing on BI, analytics and performance management-related operations and initiatives to ensure consistency and standardisation at the enterprise level and between the business units.

- **Information Management Custodians (IMC):** To coordinate and manage department level information management initiatives, decisions and implementation, and this includes data quality, business intelligence, analytics and performance metrics. Each department listed under IMCC has an IMC. An IMC is the face of his/her business unit on any information management-related matters.
- **Information Management Custodians Forum (IMCF):** The IMCF provides an environment that brings IMCs of business units to work together in resolving any issues impacting more than one business unit and make decisions to the satisfaction of the business units. This group is chaired by the senior manager of IMG.

Setting up a strong information management governance structure, without any clearly defined roles, accountabilities and responsibilities, is doomed to fail. Therefore, RACI matrix based on Control Objectives for Information Technology (COBIT) framework was used to define this. The RACI matrix helps to clearly define the roles of the different governing bodies and its members, such as business and IT managers, in the governance process. A RACI matrix consists of four attributes, namely:

- **R (Responsible):** the individual\group that performs the work and work together with A's to make decisions.
- **A (Accountable):** the individual\group that has the power to modify the business process/make decisions, is accountable for implementing the decisions and may or may not participate in discussions.
- **C (Consult):** the individual\group that is involved in discussions, but does not make decisions.
- **I (Informed):** the individual\group that needs to be informed about the outcome of the process.

A sample RACI matrix for information management is shown in Table 1 below.

## 4 Measuring Success and Rewarding Performance

To create a culture of information management in an organisation, setting up an effective strategy and governance structure alone does not suffice. Staff who contribute to the process should be recognised and rewarded for their efforts through a reward program, which would result in a positive behaviour and ultimately in a sustainable culture.

SBI General introduced ‘timely and reliable information’ as a key measure in its corporate balanced scorecard (approved by the Board) along with other performance measures. Balanced scorecard approach is used to measure performance of the

**Table 1** A sample RACI matrix

Tasks	Information asset governance factors						
<b>R (Responsible):</b> the individual\group that performs the work and works together with A's to make decisions	SBI General Board	SBI General Steering Committee	Information Management Competency Centre (IMCC)	Business Intelligence Competency Centre (BICC)	Senior executives of business units	Information management custodians (IMC)	SBI General IT
<b>A (Accountable):</b> the individual\group that has the power to modify the business process/make decisions, is accountable for implementing the decisions, may or may not participate in discussions							Strategy, Information and Performance Business Unit
<b>C (Consult):</b> the individual\group that is involved in discussions, but does not make decisions							
<b>I (Informed):</b> the individual\group that needs to be informed about the outcome of the process							
<b>Information asset governance</b>							
Accountability and ownership of information assets	A	R	R	R	R	R	R
Executive sponsorship of IMCC	A	R			R		
Endorse IM governance model, charter, roles and responsibilities	A	R	R	R	R	C	R
Executive sponsorship of BICC		R	A		R		
Coordination of IMCC and implementing the actions			R		R	R	A
Coordination of BICC and implementing the actions				I	R	R	R
Coordination of IMC Forum and implementing the actions				I		R	R
					R	A	
<b>Information asset management strategy and implementation</b>							
Produce Information Asset Management Strategic Plan for the organisation	I	I	A		C	C	R
Approve implementation of Information Asset Management Strategic Plan	I			A	R	I	R
Implement Information Asset Management Strategic Plan	I	I	R		R	R	A

organisation, departments and individuals. Because information management is a corporate measure as part of corporate strategy, this is now in the balanced scorecards of the senior executives' individual performance measures. As a result,

this measure is now cascaded to all departments to be included as part of the department balanced scorecard, which in turn is now in the department heads' balanced scorecard and, therefore, is cascaded down to employees of each department to be now part of their balanced scorecard as well. Bonuses/incentives for staffs are now tied to this measure along with other key performance indicators (KPI). This is a huge win for IM strategy—as it is part of performance measure and, therefore, is given priority by individuals and departments during the performance planning and goal setting exercise and set strategies on how they will achieve this requirement in the balanced scorecard and how they will measure success of the same.

Effective Information Quality Management (IQM) is necessary to achieve successful information management. IQM ensures that the raw material or data SBI General uses for information and knowledge is as accurate and complete as possible. Information quality management focuses on the processes, tools, principles, policies and procedures employed to maintain current, consistent and accurate information. Information Quality can only be achieved by integrating quality management principles and guidelines into the culture of the organisation.

Information assurance is the practice of managing information-related risks. More specifically, information assurance practitioners seek to protect and defend information and its supporting information systems by ensuring confidentiality, integrity, authentication, availability and non-repudiation. These goals are relevant whether the information are in storage, processing, or transit, and whether threatened by malice or accident. In other words, information assurance is the process of ensuring that authorised users have access to authorised information at the authorised time.

An information quality monitoring framework has been developed by the organisation for measuring IQM and the objectives are:

- To ascertain quality of data being captured at SBI General and reward branches, teams and employees for their efforts in capturing quality data
- To assess whether information management governance and other processes are being adhered to
- Indirectly increase awareness on importance of data quality and correct existing processes which act as deterrent to capturing quality data
- To ascertain whether the data quality message has reached all our employees at branches and that every employee in the branch is conscious of capturing quality data and its attendant benefits

Regular audit of the data in terms of quality and integrity for each department and branch through IMG and data quality and audit tools is performed using the information quality monitoring framework. A quarterly data quality and information management culture survey is circulated to all staff to measure the awareness and motivation levels among employees highlighting the data culture levels within SBI General.

Data quality measurement and reporting tool is used by the organisation to depict the quality of the data captured using some of the following key characteristics:

- Data pattern
- Data consistency
- Data correctness
- Data completeness
- Data accuracy
- Data validity

Based on the IQM review results, IMG would arrive at a score for each branch. This score would be the aggregation of the Data Culture Survey results and the Data Quality Report findings for each branch and the departments. The IQM results are made available as part of department and branch performance dashboards and corporate performance dashboards on a monthly basis. Quarterly and annual ‘data quality’ awards are given to branches and departments based on its performance against the data quality KPIs. Employees (including IMCs and data entry operators) associated with the branch or department are also rewarded with ‘Data Quality Person of the Quarter/Year’ that includes some financial incentives.

## 5 Technology Infrastructure and Support

Using the above information management best practices that were introduced at SBI General, the IT solution architecture was designed to be ‘information-centric service-driven architecture’. The IT solution is fully integrated using service-oriented architecture (SOA) principles with underlying information/data information services as the foundation to define a set of business services.

The goal of the IT architecture is to build a sustainable IT solution with information services and centricity focus. The business was actively involved in putting information-centric requirements—thanks to the IM strategic initiative that kicked off very well in advance.

### 5.1 *Consistent Language and Semantics*

Inconsistent, incomplete and inaccurate data and information with inconsistent business semantics/meaning spread across the business processes, people and supporting multiple operational business systems—such as policy, claims, customer, sales and billing—often causes insurance executives to make business decisions based on gut feel rather than analysis and is an expensive problem to fix. Insurance carriers are struggling to understand and get data in a way that enables them to do better analysis and gain some business intelligence off the data. To combat this silo approach, alleviate problems with data quality and get consistency in the way data is exchanged, shared, understood and represented, it is vital that a strategy is in place

to address this. SBI General has implemented the following programs to address this.

### 5.1.1 Enterprise Data Model

Enterprise Data Model (EDM), Canonical Model, Common Information Model (CIM), and Common Object Model (COM) are different names for the same idea: improving how business information is exchanged, understood, shared and integrated.

For SBI General, the Enterprise Data Model (EDM) was designed and developed to achieve the following objectives:

- A standard and consistent way of representing data/information for integration and exchange between the organisation business systems (includes applications) and processes.
- A standard and consistent way of representing data/information for exchange between the organisation and external partners. The partners could be government agencies and private and public organisations.

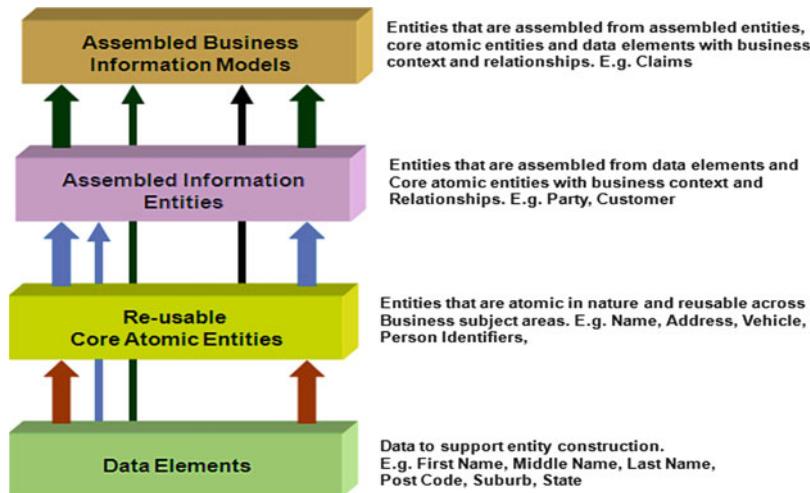
The definition of EDM for SBI General is ‘Data is defined in a consistent manner and is available for sharing among multiple business processes and the applications that support them. These common definitions are neutral with respect to the processes that produce and use the information, the applications that access the data and publish the information, and the technologies in which those applications are implemented’.

A conceptual and logical Enterprise Data Model was developed by the business that represents all business data fields whose life cycle is managed. The EDM is managed by the IMG with a clearly defined process that manages any change requests to the EDM. The physical representation of the EDM is built by IT using component model to enable reusability, and the component model framework is shown in Fig. 7.

Integration of SBI General’s IT applications is done using the principles of service-oriented architecture by using web services. The payload of web services is the EDM. The integration middleware acts as the data transformation engine that transforms application-specific data format to EDM data format and vice versa, thereby enabling loose coupling of the data structures.

### 5.1.2 Enterprise Data Dictionary

An Enterprise Data Model requires a supporting business dictionary that defines the meaning of the business terms/data fields used by it. Every entity and field in the model should be defined and described. This provides opportunities for the organisation to have a consistent understanding of the business terms used. For example, in majority of the organisations, there is no single definition of a

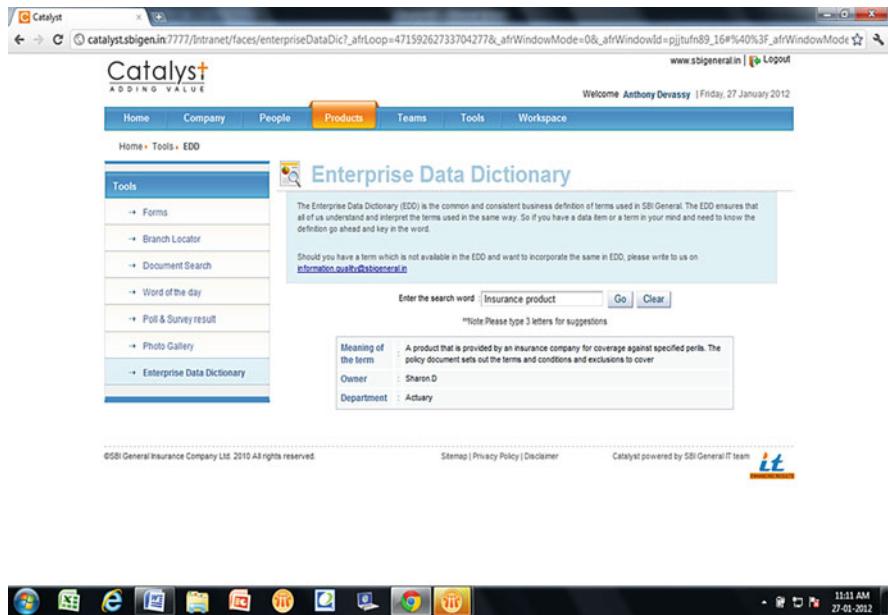


**Fig. 7** Approach to develop the EDM

‘customer’. The definition of a customer could vary depending upon the context, for example, it is different for sales, billing, marketing, underwriting and claims department. This often leads to conflicts and confusion within the organisation in terms of sharing and owning customer data. With an Enterprise Data Dictionary (EDD), these issues could be managed. An EDD at the minimum may contain the following:

- All terms in the enterprise information reference model should be defined in the dictionary.
- Each term should contain the following information:
  - Definition in business context.
  - If the term is a controlled vocabulary list, then all values should be listed with meaning.
  - Owner of the term.
  - Version number of the term with date created, who created, date modified and date deprecated.
  - How this term is defined in different business units and the meaning, and this includes the name of the term used by business units.
  - The enterprise definition of the term.
  - Source systems that capture the value for the term and the name of the term used by the source systems.

SBI General has developed an EDD. The EDD was developed by the business departments and is managed by the IMG. Changes to the EDM and EDD are managed by the IMG.



**Fig. 8** Corporate intranet portal for EDD

The EDD is published on the organisation's intranet portal so that it is available for every employee of the organisation to access it. A sample of the intranet portal is shown in Fig. 8.

## 5.2 Single Source of Truth for Key Data

In many relevant business processes, entities like customers, products, accounts, contracts and locations play a central role. These entities are known as master data, and many companies suffer today from low-quality master data scattered across the enterprise in various IT application silos. Improving master data quality and managing it more efficiently to optimise business processes is known as master data management (MDM) [4], a new term for a term known as 'single source of truth' in industry for decades. MDM is a business concept, but is widely referenced to as an IT concept as IT vendors started pushing tools to sell the MDM concept in the last couple of years. For MDM to be successfully implemented in an organisation it requires business ownership, and importantly, an MDM implementation should be seen as an enterprise-wide initiative and not just for a specific project. It may start as an implementation for a specific project, but with a plan for enterprise MDM. Many of the failed MDM projects in industry are due to lack of business ownership

and understanding of the value MDM brings to the organisation. Solid information management strategy in an organisation is critical to MDM's success.

'Dirty data' can be a problem for any type of system, but data quality and MDM are inextricably linked, because the net purpose of MDM initiatives is to deliver a single source of truth on one or more master data domains containing accurate, complete, timely and consistent data. Without early, systematic attention to high levels of data quality (plus the right data quality tools and solid data governance to resolve the issues that inevitably come up), your master data hub will simply be a fast, automated way to shoot yourself in the foot [5]. MDM should be viewed as a strategic program and not as a one off project as it manages the master assets of the organisations and therefore requires an effective and efficient governance framework that is strategic in nature and not reactive.

As more companies deploy master data management solutions, there has been an increasing demand for even more value-added capabilities from master data. One of the most significant is the demand for MDM solutions to manage enterprise-wide relationships between parties including individuals, individuals and households, individuals and corporate entities, and informal groups and organisations in one place by moving away from the silo mindset. Understanding relationships between parties and products as well as product hierarchies is critical for enterprises. This provides an organisation with a comprehensive, consistent and complete view of the relationship a party has with the organisation.

Following are some of the key principles of MDM from a 'party' perspective that were defined for SBI General:

- Every party (a person or an organisation) that SBI General deals with in some form in addition to its customers is a potential customer. So, the approach taken by SBI General was to build a single source of truth for 'party' data that supports different applications based on the role of the party (e.g. insurer, prospect, lead, agent, broker, employee, surveyor, assessor) with the organisation.
- Party data with key demographic attributes (e.g. name, address, telephone/cell number, date of birth, email address) that would help to uniquely identify or recognise the party will be stored in MDM.
- Reference data sets (e.g. list of values such as postal codes, states) that are shared by more than one application are candidates for master data.
- Relationships between parties will be managed through MDM.
- Any changes to master reference data or party data will be managed by MDM and not by applications.
- Applications can capture party data independently (e.g. quote insurance system, CRM, HR system, policy insurance system, claims management system), but MDM will store and manage the data.
- Master data will be managed and monitored by a data quality framework.
- Data specific (e.g. other than party-centric data stored in MDM) to a vertical area (e.g. general ledger, policy, claims, Human resources) is managed by the specific application and that application is the master for that data, but MDM layer is the master for party data that are shared by the application verticals.

Over 37 types of parties and over 30 reference data sets are managed by MDM at SBI General.

Three kinds of party relationships are created and managed by MDM, and they are:

- Person to person relationship (e.g. house holding)
- Person to organisation relationship (e.g. employee–employer)
- Organisation to organisation relationship (e.g. organisation–partner)

MDM is the heart of the end-to-end IT solution designed and developed by SBI General.

## 6 Conclusions and Outlook

The benefits to the business due to this information management strategic program have already started to flow in as this is being implemented. Some of the key benefits already achieved are as follows:

- As quality data is being captured in detail, the actuary is able to price our products better based on risk to the business—this is a key differentiator against our competitors as our competitors do not have a comprehensive enterprise information management strategy program implemented ground up to use and manage the data effectively and efficiently.
- There is enhanced ability to capture quality customer and potential customers in one place as the single source of truth, and thereby, we are able to recognise and understand their dealings with us through every interaction they have with us through various channels and all the products they have with us which gives us the advantage to service our customers proactively thereby providing rich and superior customer experience.
- We are able to comply with the legal and regulatory requirements from a data management perspective as we capture the required data with the right quality and hence, our reporting is effective.
- Effective corporate performance reporting in a timely manner has been made possible, with capture of accurate data for analysis that helps us in effective decision-making.
- No point-to-point integration between applications is taking place as integration is through the enterprise business data model. This helps us in quicker IT turnaround of change requests from business, better control of information flow between systems and no vendor lock-in thereby saving costs.
- Common understanding of data and information fields across the business has been established through an Enterprise Data Dictionary—a common language for data across the organisation.
- An information management group reporting to the business strategy and performance management department is established. This helps the group oversee

and manage any implications for data and information due to a business change or requirements, and managing the change through information management custodians has been ideal to be on top of data and information assets.

- There is a single place (through a Business Intelligence Competency Centre) to manage all reporting requirements thereby controlling the explosion of reports across the organisation and production and usage of reports in a consistent manner that would result in cost savings.
- A sense of ownership of data and information assets across the organisation is present due to the introduction of a solid information management governance framework with representations all departments and also by defining KPI for information asset under corporate balanced scorecard which is reflected on department and individual balanced scorecards and measured.
- Due to tight integration of data and processes, the organisation has already started to fine-tune the processes, as they are able to do so as enough data is collected to analyse and identify opportunities for improvement.
- Introduction of metrics to measure the performance of departments with regard to data and information and regular audit checks has kept the departments and its people stay focussed.
- Required data is in hand with the right quality levels to help the business to slide and dice the data to make better decisions. This is supported by a single data warehouse that collects all key data across systems in one place for effective and efficient business intelligence.
- The organisation has moved from traditional process and technology thinking to information-oriented thinking and mindset. This was a challenge for many due to lack of knowledge and skill set, but this was effectively managed through rigorous information management awareness/education sessions.

Paradoxically, doing business is becoming increasingly complex in an effort to offer customers simplicity in buying products. Some gurus say the answer is ‘customer centricity’, but what does this mean? Some say you need to reduce the number of layers between the customer and the CEO. Some put all their eggs in effective selling techniques. Others claim that marketing holds the key.

Ultimately, we all agree that the 4Ps of product, price, place and promotion remain relevant today as they were when they were originally proposed by the marketer, E. Jerome McCarthy in 1960, which has since been used by marketers throughout the world [6]. If a business promotes the right product to the right place at the right price, then the job is done. A sale in these circumstances is virtually guaranteed. This objective will satisfy all of the management buzz words including simplicity, customer centricity, effective selling and excellence in marketing.

It defies logic to suppose that a business will achieve the 4Ps without an effective approach to enterprise information management. This case study sets out one approach to achieving the foundations for effective enterprise information management. It clearly identifies the pieces of the jigsaw puzzle which are required to set a business up to effectively manage its information assets. It is a holistic view in that without all the pieces, like a jigsaw, the whole picture will not be seen.

There is no doubt that the cost savings of effective enterprise information management are substantial, but perhaps even more beneficial to any business is the long-term strategic advantages which can be gained. Access to data and information is increasing exponentially, and how this is utilised by businesses is likely to be the defining characteristic of those businesses which succeed in the medium to long term and those who fall by the wayside.

The time is ripe for businesses to take up the enterprise information management challenge as one of the most important strategic objectives for the future. We hope this chapter has provided food for thought on how to take such an objective forward.

## References

1. Radidia J (2010) Insurance Industry perspective: how insured is your data?. *Information Management Magazine*. [http://www.information-management.com/infodirect/2009\\_178/data\\_insurance\\_MDM\\_analytics\\_data\\_quality\\_security-10018858-1.html](http://www.information-management.com/infodirect/2009_178/data_insurance_MDM_analytics_data_quality_security-10018858-1.html)
2. OASIS Party Information Management Standards (OASIS Customer Information Technical Committee). OASIS. <http://www.oasis-open.org>
3. Yonke L (2008) Achieving IQ maturity: lessons learnt and best practices. Paper presented at Data Quality Asia Pacific Congress, Sydney, March 2008
4. Oberhofer M, Driebelbis A (2007) An introduction to Master Data Management Reference Architecture. IBM Developer Works. <http://www.ibm.com/developerworks/data/library/techarticle/dm-0804oberhofer/>. Accessed 17 Sept 2011
5. Power D (2012) Data quality and MDM. *Information Management Magazine*. [http://www.information-management.com/issues/2007\\_56/10014966-1.html](http://www.information-management.com/issues/2007_56/10014966-1.html)
6. McCarthy EJ (1960) Basic marketing, a managerial approach. R.D. Irwin, Homewood

# Epilogue: The Data Quality Profession

Elizabeth Pierce, John Talburt, and C. Lwanga Yonke

**Abstract** In this final chapter, we will discuss four significant topics concerning the data quality profession. First, we will examine how the data quality profession has evolved. Second, we will explore what it means to be a data quality professional. Third, we will review the training opportunities currently available to those interested in becoming a data quality professional, and finally, we will assess the outlook for the future of the data quality profession. Throughout this chapter we will use the terms “data” and “information” interchangeably.

## 1 A Brief History of the Data Quality Discipline

The information and data quality discipline has had a relatively short but rapidly evolving history that can be thought of in five phases:

- Problem Recognition: The Data Cleansing Phase
- Root Cause Detection: The Prevention Phase
- Manufacturing Analog: The Information Product and Process Management Phase
- Information Architecture: The Quality by Design Phase
- Enterprise View: Information as an Organizational Asset Phase

---

E. Pierce (✉) · J. Talburt

University of Arkansas at Little Rock, Little Rock, AR, USA  
e-mail: [expierce@ualr.edu](mailto:expierce@ualr.edu); [jrtalburt@ualr.edu](mailto:jrtalburt@ualr.edu)

C.L. Yonke

International Association for Information and Data Quality (IAIDQ), Baltimore, MD, USA  
e-mail: [lwanga.yonke@iaidq.org](mailto:lwanga.yonke@iaidq.org)

## ***1.1 Problem Recognition: The Data Cleaning Phase***

Many of the more reactive practices of data and information quality still popular today emerged as a by-product of the data warehousing movement advocated by [1,2], and others. Data warehousing was a compelling idea, but like some many great concepts, it was easier to design than to implement. The biggest impediment to data warehouse implementation turned out to be the discovery by most organizations that the data in their operational data stores was in terrible condition. It was incomplete, inconsistent, inaccurate, out of date, unreliable, and plagued by all of the other problems that we now recognize as the symptoms of poor data quality. Because these data resided in many different systems across the organization, these problems mostly lay undiscovered until there was an attempt to integrate them into a single data warehouse.

Even though Total Quality Management (TQM) was in full swing in the manufacturing and services arenas, there seemed to be a different attitude toward data. Companies that were seeking six sigma bounds on product defects seemed to be satisfied with 10 %, 20 %, or even higher levels of defects in their data stores. Redman in [3] was one of the first to expose the extent of the problem and to quantify the impact that poor data quality was having on the organization in terms of operational cost and strategic planning.

As a result, the industry of data cleansing was born, actively pursued by many organizations. The limited benefits and high costs of that approach were documented by early data quality pioneers such as English [4] and Brackett [5]. Sometimes called data cleaning or data hygiene, data cleansing focused on the use of extract-transform-load (ETL) processes to standardize the data from different sources so that it could be merged into a single data warehouse and so that queries against the data would be meaningful. To facilitate the data inspection that precedes data correction, Lindsey, Olson [6,7], and others began developing and promoting data profiling and other techniques as tools for performing data quality assessment.

## ***1.2 Root Cause Detection: The Prevention Phase***

In the next phase, data quality practitioners began to look toward the success of Total Quality Management (TQM) and to adopt some of its best practices. One of the first was root cause analysis. After the immediate need to cleanse data before building a data warehouse, organizations also began to see the value of preventing the same problems from reoccurring. Improvement projects for data quality were encouraged to put a greater emphasis on preventing future data errors first and then correcting existing data errors second [8].

### ***1.3 Manufacturing Analog: The Information Product and Process Management Phase***

Practitioners quickly realized that rather than simply adopting certain aspects of TQM, there was value in adapting the entire TQM paradigm to information, by applying manufacturing concepts to information systems [9–11] and to information processes [3, 4, 12–18].

The approach developed by Wang and his colleagues first focused on information systems, based on the view that information is the product of an information system, not a by-product [19]. By viewing data sources as raw materials, the software applications as the manufacturing process, and the final outputs as the products, then the full range of TQM principles could be applied to information systems. The result was the formulation of the Total Data Quality Management (TDQM) process [9, 19].

The approaches developed by Larry English and Tom Redman focused on defining, managing, and improving the business and IT processes through which data is created, captured, stored, delivered, used, and retired. The result was the formulation of Total Information Quality Management (TIQM) [4, 12] and the Second-Generation Data Quality Systems framework, with its specific focus on information chain management and data supplier management [17, 18].

Perhaps the most important consequence of applying the disciplines of product and process management to information is that they brought into consideration the uses and users (customers) of information. Whereas data cleansing and root cause analysis internally focused only on the data itself, with the product and process approaches, it became important to understand the information customer's perspective on the value and usefulness of the information.

### ***1.4 Information Architecture: The Quality by Design Phase***

By the time that data and information quality practices began to emerge, software development was a relatively mature process. A well-known principle of software development is that the earlier in the development process that a problem is discovered, the less effort is required to correct it. It is also reflected in Deming's 14-point plan for TQM that quality must be built in from the beginning and not inspected out at the end [20].

Through the first three phases of data cleansing, prevention, and product view, data quality practices were primarily reactive, dealing with problems and issues that were already in place in the organization. The fourth phase represents a more active role of practitioners and researchers in which they have an influence on the initial design of data models and information architectures by keeping data quality in mind from the start [4].

## 1.5 Enterprise View: Information as Organizational Asset Phase

In the fifth and current phase, there is a growing recognition of data and information as an organizational asset and resource and that data and information quality principles and practices are a critical part of maximizing the value of that asset. As a result, the focus of data quality efforts is progressively shifting from the cost side of financial statements to the revenue side [16]. As adoption of this enterprise view of data quality increases, concepts and practices identified several years ago are becoming more broadly accepted and are being refined. Perhaps one of these fundamental concepts is data stewardship, the recognition that the data are not owned by individuals or departments in an organization but that everyone is entrusted with specific responsibilities for its care and keeping (stewardship) for the good of the entire organization [4]. Closely related to data stewardship is data governance (DG), the rules and policies for making decisions about data management made by the members of a data governance council whose members represent the data stakeholders in the organization [15, 17]. A relatively newer concept is master data management (MDM), the processes, policies, and procedures around the management of the data describing employees, customers, products, facilities, equipment, and other entities that are most critical to an organization: its master data [21].

It is important to remember that these phases only represent the evolution in the understanding of data and information quality; it doesn't mean that everyone is at the same level. Quite the contrary, just as the Capability and Maturity Model (CMM) for software development defines five levels of maturity, most organizations are still operating at Levels 1 or 2. Similarly, the data quality programs in most organizations are still focused on basic data cleansing with perhaps some root cause analysis. Few have yet to put into place a comprehensive enterprise-wide data quality program incorporating the components that comprise the higher levels of data and information quality [12, 16, 22].

## 1.6 What's Next?

The next logical step in the evolution of the data and information quality is an expansion of the current paradigm from a single enterprise to multiple organizations. Movement in this direction has already been signaled by attempts at creating various data standards such as XBRL (eXtensible Business Reporting Language) for the finance industry and the APCD (All-Payer Claims Database) and NCPDP (National Council for Prescription Drug Programs) standards for healthcare. Another is the ISO 8000–110: 2009 on Master data: Exchange of characteristic data: Syntax, semantic encoding, and conformance to data specification. All are focused on either standardizing or establishing rules for how to standardize information shared between organizations. Data quality will also need to evolve to ensure it

appropriately addresses issues arising from new data create and use processes, such as social media.

## 2 The Job of the Data Quality Professional

As the data quality field has evolved, so too have the roles, responsibilities, and training of individuals engaged in the pursuit of better data for their organizations. As part of its efforts to better understand the data quality profession and to develop standards for data quality professionals, the International Association for Information and Data Quality (IAIDQ) has sponsored several important studies. The first one was a role delineation/job analysis study conducted between October 2008 and March 2009, to serve as the basis for the Information Quality Certified Professional (IQCP<sup>SM</sup>) credential developed by the International Association for Information and Data Quality, IAIDQ (<http://iaidq.org>) [23,24]. This study extended work previously done by several others [13, 25, 26]. The second study was the very first salary and job satisfaction survey for the data quality profession, conducted in 2009 by a team of investigators from IAIDQ and the University of Arkansas at Little Rock [27]. That 2009 survey also included questions about how data quality professionals spent their time at work, what characterized their work environment, and where they obtained their educational backgrounds. Much of the material for this section of the chapter originates from those two studies.

### 2.1 *Recognizing the Data Quality Professional*

Unlike more mature fields such as accounting, law, or medicine, there are no consistent degrees, job titles, or state license boards to rely upon when it comes to identifying a data quality professional. Of the 120 job titles self-reported by individuals engaged in data quality activities in IAIDQ's 2009 survey, fewer than 30 % contained either the phrase "Information Quality" or "Data Quality." About 40 % of the titles contained only the word "Information" or "Data" but no reference to "Quality." Of the remaining titles, 2.5 % included the word "Quality" (but not "Information" or "Data"), and 27.5 % have no reference to any of the words "Information," "Data," or "Quality."

To complicate matters further, activities associated with the creation, management, consumption, and improvement of data are performed by many individuals in an organization, not just a few dedicated specialists. Moreover, even those charged with data quality-related duties may have other job responsibilities not related to data quality.

As a result of these difficulties, for certification purposes, IAIDQ defined data quality professionals as people who "hold any of a wide range of positions in their organizations, as individual contributors or as managers. They conduct, lead, champion or participate in information quality projects. They work in any of

the functions or disciplines within their organization or are part of a specialized information quality team; yet all perform information quality activities as part of their job responsibilities. This information quality work is either part-time within a broader organizational role, or on a full-time basis” [23].

The 2009 salary and job satisfaction survey provided further insights on how data quality professionals allocate their time across the six performance domains. The six domains are listed here in order of decreasing time spent.

### **2.1.1 Information Quality Measurement and Improvement Domain**

The Information Quality Measurement and Improvement domain covers the steps involved in executing data quality projects. Activities include gathering and analyzing business requirements for data, assessing the quality of data, determining the root causes of data quality issues, developing and implementing information quality improvement plans, preventing and correcting data errors, and implementing information quality controls. Of all the domains, this one was cited as the set of work most frequently performed by individuals responding to IAIDQ 2009 salary and job satisfaction survey [27].

### **2.1.2 Information Architecture Quality Domain**

The Information Architecture Quality domain is comprised of the tasks that assure the quality of the data blueprint for an organization. Activities include participating in the establishment of data definitions, standards, and business rules; testing the quality of the information architecture to identify concerns; leading improvement efforts to increase the stability, flexibility, and reuse of the information architecture; and coordinating the management of metadata and reference data. This domain along with the next (Sustaining Information Quality domain) was the second set of work most frequently performed by individuals responding to IAIDQ 2009 salary and job satisfaction survey [27].

### **2.1.3 Sustaining Information Quality Domain**

The Sustaining Information Quality domain focuses on putting in place processes and management systems that ensure ongoing information quality. Duties include acting as an information quality consultant for integrating data quality activities into other projects and processes (e.g., data conversion and migration projects, business intelligence projects, customer data integration projects, enterprise resource planning initiatives, or system development life cycle processes) and continuously monitoring and reporting data quality levels. This domain along with the previous (Information Quality Architecture domain) ranked second in frequency of performance by individuals responding to IAIDQ 2009 salary and job satisfaction survey [27].

### **2.1.4 Information Quality Strategy and Governance Domain**

The Information Quality Strategy and Governance domain includes the efforts that provide the structures and processes for making decisions about an organization's data as well as ensuring that the appropriate people are engaged to manage information throughout its life cycle. Activities include working with key stakeholders to define and implement information quality principles, policies, and strategies; organizing data governance by naming key roles and responsibilities; establishing decision rights; and building essential relationships with senior leaders in order to improve information quality. This domain along with the next (Information Quality Value and Business Impact domain) was the third set of work most frequently performed by individuals responding to IAIDQ 2009 salary and job satisfaction survey [27].

### **2.1.5 Information Quality Value and Business Impact Domain**

The Information Quality Value and Business Impact domain consists of the techniques used to determine the effects of data quality on the business as well as the methods for prioritizing information quality projects. Activities include evaluating information quality and business issues, prioritizing information quality initiatives, obtaining decisions on information quality projects, and reporting results to demonstrate the value of information quality improvement to the organization. This domain along with the previous (Information Quality Strategy and Governance domain) ranked third in frequency of performance by individuals responding to IAIDQ 2009 salary and job satisfaction survey [27].

### **2.1.6 Information Quality Environment and Culture Domain**

The Information Quality Environment and Culture domain provides the background that enables an organization's employees to continuously identify, design, develop, produce, deliver, and support the information quality that customers need. Activities include designing information quality education and training programs, identifying career paths, establishing incentives and controls, promoting information quality as part of business operations, and fostering collaborations across the organization for the purpose of engaging people at all levels in information quality strategies, principles, and practices. This domain was the least frequently performed by individuals responding to IAIDQ 2009 salary and job satisfaction survey [27].

### **2.1.7 Career Paths for Data Quality Professionals**

The 2009 salary and job satisfaction survey also shed light on career paths for data quality professionals. In terms of their career experiences, data quality professionals can be found around the globe, working in nearly every industry with

**Table 1** Industries employing data quality professionals

Which of the following best describes your organization's primary industry? (n = 115)		
Category	Number of responses	Percentage of responses
Banking/financial services	18	16.5 %
Consulting/professional services	14	12.8 %
Insurance	11	10.1 %
Retail/wholesale distribution	8	7.3 %
Energy/oil and gas/utilities	8	7.3 %
Healthcare	8	7.3 %
Government: state/local	7	6.4 %
Government: federal/national	6	5.5 %
Manufacturing	6	5.5 %
Telecommunications/communications	6	5.5 %
Aerospace	4	3.7 %
Software/Internet	3	2.8 %
Education	2	1.8 %
Food/beverage/agriculture	2	1.8 %
Biotechnology/pharmaceuticals	1	0.9 %
Advertising/marketing/public relations	1	0.9 %
Chemical	1	0.9 %
Law	1	0.9 %
Logistics/transportation	1	0.9 %
Nonprofit/trade association	1	0.9 %

banking/financial services (16.5 %), consulting/professional services (12.8 %), and insurance (10.1 %) as the most frequently cited. Industries like banking, financial services, and insurance handle huge amounts of data that are subject to stringent regulations regarding the accuracy and privacy of that data. This provides these firms with a strong incentive for hiring individuals with specialized expertise in data quality. While some of this expertise comes from in-house sources, engaging an external consulting/professional services firm is another avenue for organizations to acquire the data quality expertise that they need. In addition to the industries listed in Table 1, other industries employing data quality professionals contributed by IAIDQ survey participants include IT Services (e.g., Systems Integration/Outsourcing, Computing and Services), Audit/Consulting Services, Consumer Product Goods, and Research (e.g., scientific surveys, clinical trials) [27].

Most data quality professionals report working full time for either public companies (43.6 %) or private companies (32.5 %). Another 9.4 % are employed by nonprofit organizations, and 14.5 % have jobs with some government agency. Data quality professionals typically work for larger organizations with nearly a third responding that they work for organizations with 50,000 or more employees (31.6 %). This is further borne out by the revenue size of organizations with almost a third of them estimating their organization's 2008 annual revenues to be more than \$10 billion (30.4 %) [27]. Although good-quality data is of value to organizations of any size, it is typically the larger organizations that currently have the resources and

**Table 2** Departments housing data quality professionals

In which department is your position located? (n = 105)		
Category	Number of responses	Percentage of responses
Information Technology/information Systems (IT/IS)	60	57.1 %
Accounting/finance	11	10.5 %
Marketing/sales	6	5.7 %
Audit/compliance/risk	5	4.8 %
Production/operations/maintenance	5	4.8 %
Quality assurance	4	3.8 %
Design/engineering	2	1.9 %
Research and development	2	1.9 %
Human resource	2	1.9 %
Purchasing/supply chain management	2	1.9 %
Legal	1	1.0 %
Process management	1	1.0 %
Not applicable: I am self-employed	4	3.8 %

economies of scale to hire individuals specifically designated for data governance and stewardship roles.

In terms of where data quality professionals are placed in an organization, over half of the data quality professionals surveyed (57.1 %) said their position is located within an Information Technology/Information Systems department (IT/IS). This in turn means that 42.9 % of the survey respondents do not work in IT/IS. This clearly dispels the myth held by many that data quality is solely a responsibility of the IT/IS discipline.

Among the non-IT/IS departments, accounting/finance (10.5 %), Marketing/Sales (5.7 %), Audit/Compliance/Risk (4.8 %), and Production/Operations/Maintenance (4.8 %) were the most common business areas that employed data quality professionals. As of yet, very few organizations have a unit devoted entirely to data quality activities. A review of Table 2 summarizing the entire collection of survey responses revealed that although the IT/IS function was named most often, the range of areas where data quality professionals are employed is extensive, encompassing nearly all parts of an organization. This is an indication that to be successful, data stewardship must be a shared responsibility between the IT/IS group and all the organization's business units [27].

### 3 Training for Data Quality Professionals

Given the wide range of roles and responsibilities covered by the six domains identified by IAIDQ, how does one prepare for a career devoted to the improvement of data quality? Because data quality concepts and techniques were developed by

industry practitioners, until recently, there has not been a widely accepted body of knowledge and a common vocabulary for describing a skill set for data quality. Whereas a discipline like computer science has a long history of academic research and publication in topics such as algorithm design, the theory of computation, and proof of correctness for algorithms, the same cannot be said for the data quality field. However, this is beginning to change as more conferences and journals are soliciting and publishing articles in this area. Recent papers by Ge and Helfert and Madnick et al. [28,29] describe the emerging framework of data/information quality research. In the practitioner community, English [4] identified typical data quality training topics for various organizational roles. Consequently there is a gradual movement of both academic and professional programs to develop training geared specifically towards providing individuals engaged in data quality activities with the knowledge areas and skills that they need.

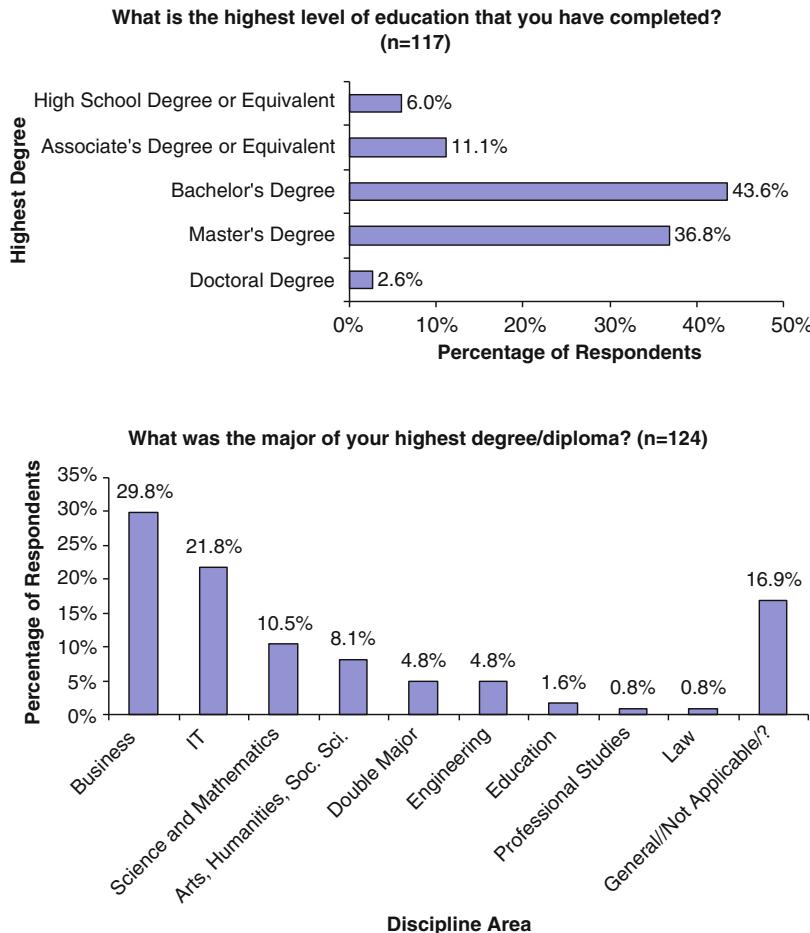
Based on the findings of the 2009 IAIDQ salary and job satisfaction report, most data quality professionals start with obtaining a Bachelor's or Master's degree. While the disciplines vary, business (29.8 %) and IT-/IS-related areas (21.8 %) are the most common academic preparation [27] (Fig. 1).

How have today's data quality professionals developed and learned the skills they need for the job? Typically, once an aspiring data quality professional has gained some mastery of a particular subject area (e.g., business, technology, science, or mathematics) along with the IT/IS techniques for managing the data encompassed by that subject, the next step is to augment that knowledge with a combination of self-study, specialized courses, and/or professional training in data quality concepts and best practices. A recent survey indicates that the overwhelming majority of data quality professionals did not receive any formal training in data quality management. Sixty-three percent (63 %) of respondents indicated that they were self-taught, which often was combined with on-the-job training (47 %). Twenty-four percent (24 %) received professional training related to data quality, and 20 % reported having a university training related to data quality [30]. The survey by Sadiq et al. [30] also noted the significant negative impact that the lack of standardized or best-practice data quality education has on the quality, consistency, sustainability, and eventual success of data quality management as currently practiced.

Two recent developments have become significant milestones in the efforts to define standards for data quality knowledge and skills and education:

- The introduction of data quality courses at colleges and universities worldwide, culminating in the establishment of the Information Quality Program at the University of Arkansas at Little Rock
- The introduction of the Information Quality Certified Professional (IQCP) credential, by IAIDQ

We briefly discuss the IQCP credential here and follow with an overview of data quality academic programs and others sources of data quality education and training.



**Fig. 1** 2009 IAIDQ salary and job satisfaction report [27]

### 3.1 International Professional Certification: The IAIDQ Information Quality Certified Professional (IQCP) Credential

Chartered in 2004, IAIDQ is the only international professional organization devoted entirely to information and data quality. It has members in more than 30 countries on five continents. In February 2011, IAIDQ introduced the Information Quality Certified Professional (IQCP) credential. It is rapidly becoming the global standard of competence for data quality practitioners. The IQCP certification has

three components (IAIDQ; <http://iaidq.org>):

- Work experience and education requirements
- Taking and passing a comprehensive 3-h exam consisting of 150 multiple choice questions with four possible answers each. Following Bloom's Taxonomy, the questions assess three cognitive domains: Recall/Understanding, Application, and Analysis
- Signing the IAIQD Code of Ethics and Professional Conduct

The IQCP credential must be renewed every 3 years. There are two ways to recertify:

- Submit a recertification journal that documents a minimum level of ongoing professional development
- Or take the exam again

The certification is based on the findings of a job analysis/role delineation study sponsored by IAIQD between October 2008 and March 2009 and conducted with CASTLE Worldwide, Inc. (CASTLE), a firm that specializes in the development of professional certifications [23, 24]. The purpose of the study was to build IAIQD's Information Quality Certified Professional credential on a solid foundation validated by practitioners and consistent with best practices. The process followed by IAIQD and CASTLE complies with widely accepted standards and regulations, such as the ISO/IEC 17024 for Personnel Certification Bodies. The exam content is independent of any specific methodology, vendor, or tool.

The panel of experts assembled for the job analysis/role delineation study and developed a consensus definition of the job of the IQCP which consists of a framework containing six (6) performance domains, twenty-nine (29) tasks, and 207 distinct knowledge and skills. After it was validated by a large international group of information/data quality practitioners, the framework was used to develop the specifications for the IQCP Exam. The six performance domains are:

- Information quality strategy and governance
- Information quality environment and culture
- Information quality value and business impact
- Information architecture quality
- Information quality measurement and improvement
- Sustaining information quality

The 207 distinct knowledge areas and skills are further classified into the following five groups:

- IQ/DQ core
- Quality foundation
- Leading the IQ/DQ effort
- Information management
- People and interpersonal effectiveness

Beyond this primary purpose as the blueprint for the certification exam, the IQCP Framework is also expected to:

- Drive an increase in the quality and consistency of the information/data quality training available in the market place
- Provide a benchmark against which organizations can assess their information/data quality practices

As another indicator of the need for formal data quality training and best practice data quality education standards and benchmarks, Yonke et al. [23] found 43 % of respondents that they surveyed indicated that the most important benefit they expected from a data quality certification was “increased knowledge and mastery of the information/data quality discipline” [23].

### ***3.2 Data Quality Education and Training Sources***

In addition to opportunities for self-study through published sources (e.g., social media, books, and journal articles), several formal educational opportunities in data quality are available. These offerings can be categorized as follows:

- The MIT Information Quality Program
- The UALR Information Quality Program
- Data Quality Education at Other Colleges and Universities
- Information Quality Communities of Practice and Certification
- Industry Conferences and Practitioner-Provided Training

#### ***3.2.1 The MIT Information Quality Program***

Much of the credit for introducing academic rigor into the field of data and information quality belongs to the Massachusetts Institute of Technology (MIT) information quality program. Responding to industry needs for high-quality data and inspired by the success of the Total Quality Management movement in manufacturing, Dr. Stuart Madnick in the MIT Sloan School of Management led a partnership of organizations to create a research program in the early 1990s called Total Data Quality Management (TDQM). While the short-term focus of TDQM was to create a center of excellence among practitioners of data quality techniques, its greatest impact has been to build an academic research community to investigate the theory of data and information quality as well as documenting its best practices.

One of the most successful products of the TDQM program is the MIT information quality program (MITIQ) led by Dr. Richard Wang and housed in the MIT Center for Technology, Policy, and Industrial Development (CTPID). The MITIQ program has been the leader in promoting and disseminating research in

information quality through its sponsorship of the International Conference on Information Quality (ICIQ). The ICIQ has been held annually since 1996 and has created a worldwide community of academicians and practitioners who regularly present at the conference and publish their peer-reviewed papers in its proceedings.

The members of the MITIQ community have been active in developing many of the fundamental principles and tenets of information quality. Examples of these include the studies on impact of poor information quality [3, 31], the dimensions of data quality [11], information as product [10], and information quality assessment and improvement methodologies [32].

Many key initiatives have also had their roots in the MITIQ community as well. One of the most important was the 2009 launch of the Association for Computing Machinery (ACM) Journal of Data and Information Quality (JDIQ) with Dr. Madnick and Dr. Yang Lee (Northeastern University), as its Founding Editors-in-Chief. Another was the organization of the annual Information Quality Industry Symposium (IQIS) established in July 2007. Originally begun to complement the research focus of the ICIQ, the IQIS conference was intended to promote the sharing of best practices and technology among IQ practitioners, IQ tool vendors, and professional organizations promoting IQ. The most recent initiative has been the MIT Chief Data Officer (CDO) Forum started in 2011. Meeting in conjunction with the July IQIS, the new event has been reformulated as the Chief Data Officer and Information Quality (CDOIQ) Symposium to reflect its emphasis on establishing information quality as an organizational function led at the enterprise level by the Chief Data Officer (CDO Forum, 2012).<sup>1</sup>

### 3.2.2 The UALR Information Quality Graduate Program<sup>2</sup>

Despite the rapid growth in information quality practices that arose out of the data warehousing during 1990s, there was not a corresponding growth in academic programs. In 2000, Craig Fisher introduced an undergraduate course titled Data Quality and Information Systems at Marist College [33]. This subsequently led to the publication of the first college-level textbook on IQ, Introduction to Information Quality [34].

In 2005, Dr. Richard Wang working in collaboration with Charles Morgan, the company leader of Acxiom Corporation headquartered in Little Rock, Arkansas, and with Dr. Mary Good, Dean of the Donaghey College of Engineering and Information Technology (EIT) at the University of Arkansas at Little Rock (UALR), conceived a plan to create the first graduate degree program in information quality [35]. A Master of Science in Information Quality (MSIQ) was the first program developed

---

<sup>1</sup>CDO Forum (2012) The Second MIT Chief Data Officer Forum (<http://mitiq.mit.edu/CDO/2012/>).

<sup>2</sup>The University of Arkansas at Little Rock Information Quality Graduate Program, UALR IQ (<http://ualr.edu/informationquality/>).

**Table 3** Information quality course offerings at the University of Arkansas at Little Rock

PhD in Integrated Computing with emphasis in Information Quality	INFQ 7303 Principles of Information Quality INFQ 7342 Information Quality Tools and Industry Landscape INFQ 7367 Information Quality Policy and Strategy 7318 Total Quality Management and Statistical Process Control INFQ 7337 Project and Change Management
Master's degree in Information Quality	INFQ 7322 Information Quality Theory INFQ 7348 Entity Resolution and Information Quality IFSC 7360 Data Protection and Privacy IFSC 5330 Database Security
Graduate Certificate in Information Quality	IFSC 5345 Information Visualization IFSC 7320 Database Systems and Information Architecture IFSC 7310 Information Systems Analysis

and approved, and the first cohort of 24 students was enrolled in the fall of 2006. The UALR Information Quality Graduate Program (UALR IQ, 2012) is housed in the Department of Information Science in the Donaghey College of Engineering and Information Technology and has expanded to include a Graduate Certificate in Information Quality and an information quality emphasis track in the Integrated Computing PhD program (Table 3).

The curriculum development for the program was a joint effort among several collaborators [36]. Much of the course content has to be developed from the ground up including the Principles of Information Quality course [37], the Information Quality Tools course [38], and Entity Resolution and Information Quality [36].

As the only university program granting graduate degrees in information quality in the United States, it was decided from its inception that the program should have a distance education component. Beginning in 2007, the program was made available online by live webcasting of the classes required for the MSIQ program. Unlike traditional online classes where students work at their own pace, the UALR IQ online program is synchronized with the on-campus course offerings. Each course has an on-campus classroom and normal meeting schedule. The classrooms are specially equipped so that as lectures are being delivered to the local students, they are also webcast to remote students in realtime. The webcast system allows remote students to see whatever is displayed on the instructor's desktop and to have both chat and audio interactions with the instructor and other members of the class. The webcasts are also recorded for later viewing. Remote students are required to take their major examinations through a test proctoring service [39].

Students across the USA in states such as California, Texas, New York, Georgia, and North Carolina, and in several foreign countries such as Brazil and South Africa have successfully completed their degrees online. As of May 2012, the UALR IQ

**Table 4** Colleges and universities offering IQ/DQ courses

School	Program name	Courses
Marist College, Poughkeepsie, New York, USA	BS w/ Major in Information Technology and Systems MS: MSIS degree	ITS 428 Data Quality in Information Systems MSIS 557 Data Quality in Information Systems
University of Nebraska at Omaha, Omaha, Nebraska, USA	Information Systems Quantitative Analysis	Information Quality and Data Management (ISQA 8206/4200)
Northeastern University, Boston, Massachusetts, USA	MBA MIS/Honors Undergraduates	Information Quality: Technology and Philosophy Information Quality for Global Managers Information Resource Management
The University Of Michigan-Dearborn, Dearborn, Michigan, USA	MSE in Manufacturing Systems Engineering	IMSE 532 - Information for Manufacturing
Dublin City University, Dublin, Ireland	MSc in Business Informatics	Research at the Business Informatics Group
Nyenrode Business University, Breukelen, Netherlands	Strategic Business Planning	Masterclass Data Quality Management
University of South Australia, Adelaide, SA, Australia	Doctor of Information Science	Research Program
University of St. Gallen, St. Gallen, Switzerland	Competence Center Corporate Data Quality	Consortium Research Program
University of Westminster, London, UK	MSc Information Quality	Course modules in Information Quality, Information Security, Data Warehousing and Data Mining, Enterprise Applications, Statistical Modeling, Postgraduate Project

program has graduated 58 students with the MSIQ degree and 7 students with the Integrated Computing PhD degree with an Information Quality emphasis.

### 3.2.3 Data Quality Education at Other Colleges and Universities

Since 2006, several other colleges and universities have developed graduate-level programs in data quality (Table 4). These include a Master of Science in Information Quality at the University of Westminster in London, UK, and Graduate Studies in Information Quality at the University of South Australia in Adelaide. In addition, although not offering full degree programs, many schools have introduced courses in data quality into existing business or IT degree plans. IAIDQ maintains an updated list on its website (IAIDQ; <http://iaidq.org>).

### 3.2.4 Information Quality Communities of Practice

Information Quality Communities of Practice are groups organized on a volunteer basis by data quality professionals to promote awareness of the discipline and to better share data and information quality best practices. The major Information Quality Communities of Practice include the international association IAIDQ and several national organizations such as DGIQ in Germany, ExIQ in France, AECDI in Spain, ArgIQ in Argentina, and QIBRAS in Brazil. These communities of practice are a valuable resource for sharing experiences and disseminating knowledge regarding data quality techniques and best practices.

In November 2010 at the 15th International Conference for Information Quality held at the University of Arkansas at Little Rock, representatives of the major national Information Quality Communities of Practice from around the world agreed to upgrade their current course materials so they could help their members prepare for the Information Quality Certified Professional exam. In addition to generating these education materials, they also plan to develop a special version for universities in order to provide a worldwide education base of high-quality instructional units.

### 3.2.5 Industry Conferences and Practitioner-Provided Training

This last group is probably responsible for the bulk of data quality training available thus far. Combining half-day or full-day tutorials with conference presentations and case studies, practitioner-led international conferences held around the world have played a key role in developing data quality professionals. We note in particular:

- The Information Quality Conference organized by Larry English and held in the USA between 1999 and 2005.
- The conferences organized by IAIDQ and its partners and held in the USA since 2006 ([www.idq-conference.com](http://www.idq-conference.com)).
- The Data Management and Information Quality Conference Europe held in London since 1999 and organized by IRM UK and its partners including Larry English and DAMA ([www.irmuk.co.uk](http://www.irmuk.co.uk)).
- Data Quality conference in Sydney, Australia, organized by Ark Group Australia since 2006, and under the brand name Data Quality Asia Pacific Congress since 2008 ([www.dqasiapacific.com](http://www.dqasiapacific.com)).

Practitioner training focusing on data and/or information quality is also offered from various consultants, professional organizations, software vendors, training institutes, and government agencies. Software vendors often link their courses closely to their own products that they offer for resolving data quality problems, while training institutes use their data quality modules to enhance their education program portfolios. Some examples of organizations offering data quality training are listed in Table 5.

**Table 5** Examples of organizations offering data quality training and consulting

Organization	URL
Information Impact International, Inc.	<a href="http://www.infoimpact.com/">http://www.infoimpact.com/</a>
Knowledge Integrity, Inc.	<a href="http://knowledge-integrity.com/">http://knowledge-integrity.com/</a>
Granite Falls Consulting	<a href="http://www.gfalls.com/index.html">http://www.gfalls.com/index.html</a>
Navesink Consulting Group	<a href="http://www.dataqualitysolutions.com/">http://www.dataqualitysolutions.com/</a>
The Data Governance Institute	<a href="http://www.datagovernance.com/">http://www.datagovernance.com/</a>
Institute for Certification and Training of Information Professionals	<a href="http://ictip.org">http://ictip.org</a>
Castlebridge Associates	<a href="http://castlebridge.ie">http://castlebridge.ie</a>
eLearningCurve	<a href="http://ecm.elearningcurve.com">http://ecm.elearningcurve.com</a>
BI Community	<a href="http://www.bi-community.org">http://www.bi-community.org</a>

## 4 The Future for the Data Quality Profession

The data quality profession continues to grow. Although costs and revenue issues associated with poor-quality data have always been a factor, it appears that compliance and regulatory issues have become a major driver for organizations to invest in treating their information as a strategic asset [23, 40]. This motivation stems from regulation specifically aimed at improving the quality of data as well as consequences of privacy legislation. The Data Protection Act in 1998, the Federal Data Quality Act in 2001, the Sarbanes-Oxley Act in 2002, and Basel II in 2004 all contain language regarding better data management so as to ensure the accuracy of the data being reported to the Federal Government. Privacy legislation such as the National Do Not Call Registry and HIPAA has made it imperative that organizations maintain good-quality data regarding their customers in order to meet their legal obligations. Issues such as these along with the growth of data warehousing, business intelligence, and master data management initiatives have spurred organizations' desire for better data.

Today more organizations than ever are moving towards treating their data as critical assets that must be effectively governed for maximum value. To do this, organizations need everyone who works with data whether they are a data creator, data processor, or data consumer to appreciate the basic tenets of the six domains that IAIDQ has defined as key data quality knowledge areas. In addition, organizations require individuals with more in-depth training and understanding to take on leadership roles in implementing data quality best practices as part of the organization's day-to-day data management and stewardship activities. Universities, communities of practice, and professional trainers/consultants all have a role to play in providing educational opportunities. Web resources in the form of websites, Web training videos, white papers, wikis, and electronic serial collections make it economically feasible to disseminate tools, techniques, and lessons learned on a global scale.

Looking forward, the main challenge for data quality professionals will be overcoming the obstacles that prevent organizations from maximizing the value of their information. A 2012 survey conducted by UALR and IAIDQ revealed that data

quality professionals continue to face numerous challenges in their organizations. These obstacles include:

- Lack of accountability and responsibility for data quality
- Too many information silos
- Lack of awareness or communication of the magnitude of data quality problems
- Lack of common understanding of what data quality means
- Lack of awareness or communication of the opportunities associated with high-quality data.
- Lack of senior leadership in tackling data quality issues.
- Lack of data quality policies, plans, and procedures.
- Perception that data quality is an IT issue only rather than an organization-wide issue, and in some organizations, there may be a reverse perception that data quality is a business issue only and cannot be helped with IT support.
- Lack of data quality goal setting and measurement.
- Lack of data quality skills and expertise.
- Lack of data quality tools and automation.
- Lack of resources including limited staff to manage data issues and promote data quality, cost to build a good data quality program, and time to get proper tools and automation in place.
- Out of date policies, plans, and procedures.
- Lack of grass roots development of data quality as a strategic vision.
- Lack of data quality rules that are customer focused.
- Lack of understanding by data collectors of their impact on quality.
- Lack of awareness of impact of frequent organizational changes on contextual meaning and usability of data assets.

If data and information quality is to make progress as a discipline, these obstacles must be alleviated. Many of these problems have at their root a lack of awareness by organizational leaders of either the negative impacts of poor data quality or a lack of awareness that there exist effective strategies and practices for eliminating poor data quality [12, 16, 41]. Communities of practice, academic centers, and leading practitioners must continue their efforts to spread the message while they build upon the data quality knowledge base. By combining traditional learning avenues such as university and college courses and degrees, conferences and workshops with the power of online courses and social media such as webinars and wikis, individuals in the data quality field can help provide a rich infrastructure for those trying to establish and grow a culture in their organization that supports and promotes data quality improvement.

## References

1. Inmon WH (1992) Building the data warehouse. Wiley, New York
2. Kimball R, Ross M, Thornthwaite W, Mundy J, Becker B (1998) The data warehouse lifecycle toolkit. Wiley, New York

3. Redman TC (1998) The impact of poor data quality on the typical enterprise. *Commun ACM* 41(2):79–82
4. English LP (1999) Improving data warehouse and business information quality: methods for reducing costs and increasing profits. Wiley, New York
5. Brackett M (1996) The data warehouse challenge: taming the chaos. Wiley Computer Publishing, New York
6. Lindsey E (2008) Three-dimensional analysis: data profiling techniques examining data content, structure, and quality. Data Profiling LLC, USA
7. Olson J (2003) Data quality: the accuracy dimension. Morgan Kaufmann, Burlington
8. McGilvray D (2008) Executing data quality projects: ten steps to quality data and trusted information. Morgan Kaufmann, Burlington
9. Wang RY, Kon H-B (1993) Toward Total Data Quality Management (TDQM). In: Wang RY (ed) *Information technology in action: trends and perspectives*, 1st edn, Prentice Hall, Englewood Cliffs
10. Wang RY, Lee YW, Pipino LL, Strong DM (1998) Manage your information as a product. *Sloan Manag Rev* 1998(Summer):95–105
11. Wang RY, Strong DM (1996) Beyond accuracy: what data quality means to consumers. *J Manag Inf Syst* 12(4):5–34
12. English L (2009) *Information quality applied: best practices for improving business information, processes, and systems*. Wiley, New York
13. English LP (2008) Information quality management: job position roles. IDQ Newsletter 4(2), International Association for Information and Data Quality, IAIDQ. <http://iaidq.org/publications/english-2008-04.shtml>
14. Redman TC (2007) The body has a heart and soul: roles and responsibilities of the Chief Data Officer. IDQ Newsletter 3(1), International Association for Information and Data Quality, IAIDQ. <http://iaidq.org/publications/redman-2007-01.shtml>
15. Redman TC (2005) A comprehensive approach to data quality governance. In: A Navesink Consulting Group White Paper. International Association for Information and Data Quality, IAIDQ. <http://iaidq.org/publications/redman-2005-06.shtml>
16. Redman TC (2008) *Data driven: profiting from your most important business asset*. Harvard Business Press, Boston
17. Redman TC (2001) *Data quality: the field guide*. Digital Press, Surry Hills
18. Redman TC (1997) *Data quality for the information age*. Artech House, Norwood
19. Huang K-T, Lee YW, Wang RY (1999) *Quality information and knowledge*. Prentice Hall, Upper Saddle River
20. Deming WE (1986) *Out of the crisis*. MIT Press, Cambridge
21. Loshin D (2008) *Master data management*. Morgan Kaufmann, Burlington
22. Baskarada S (2009) Information quality management capability maturity model. Vieweg+Teubner, Germany
23. Yonke CL, Walenta C, Talburt JR (2011) The job of the information/data quality professional. Published by IAIDQ. <http://iaidq.org/publications/yonke-2011-02.shtml>
24. Yonke CL, Walenta C, Talburt JR (2011) The skills of the information/data quality professional. International Association for Information and Data Quality, IAIDQ. <http://iaidq.org/publications/yonke-2011-11.shtml>
25. Chung WY, Fisher CW, Wang R (2002) What skills matter in data quality? In: Proceedings of the 7th international conference on information quality, MIT, Cambridge, 8–10 November 2002. MIT IQ Publishing, Cambridge
26. Pierce EM (2003) Pursuing a career in information quality: the job of the data quality analyst. In: Proceedings of the 8th international conference on information quality, MIT, Cambridge, 7–9 November 2003. MIT IQ Publishing, Cambridge
27. Pierce EM, Yonke CL, Lintag A (2009) 2009 Information/Data Quality Salary and Job Satisfaction Report: Understanding the Compensation and Outlook of Information/Data Quality Professionals. International Association for Information and Data Quality, IAIDQ. <http://iaidq.org/publications/pierce-2009-07.shtml>

28. Ge H, Helfert H (2006) A review of information quality research. In: Proceedings of the 11th international conference on information quality, MIT, Cambridge, 10–12 November 2006. MIT IQ Publishing, Cambridge
29. Madnick SE, Wang RY, Lee Y-L, Zhu H (2009) Overview and framework for data and information quality research. *J Data Inf Qual* 1(1):2.1–2.22
30. Sadiq S, Indulska M, Jayawardene V (2011) Research and industry synergies in data quality management. In: Proceedings of the 16th international conference on information quality, MIT, Cambridge, 18–20 November 2011. MIT IQ Publishing, Cambridge
31. Fisher CW, Kingma BR (2001) Criticality of data quality as exemplified in two disasters. *Inf Manag* 39:109–116
32. Pipino L, Lee Y, Wang R (2002) Data quality assessment. *Commun ACM* 45(2):211–218
33. Fisher CW (2001) A college course: data quality in information systems. In: Proceedings of the 6th international conference on information quality, MIT, Cambridge, 2–4 November 2001. MIT IQ Publishing, Cambridge
34. Fisher C, Lauria E, Chengalur-Smith S, Wang R (2006) Introduction to information quality. M.I.T. Information Quality Program, Cambridge
35. Talburt J (2011) Entity resolution and information quality. Morgan Kaufmann, Burlington
36. Lee Y, Pierce E, Talburt J, Wang R, Zhu H (2007) A curriculum for Master's of Science in Information Quality. *J Inf Syst Ed* 18(2):233–242
37. Zhou Y, Talburt JR (2011) A gateway course for an information quality graduate program. In: Proceedings of the international symposium on information systems and management in Asia Pacific, ISMAP-2011, Xi'an, China, 9–11 December 2011
38. Zhou Y, Talburt JR (2011) Developing a graduate-level course on information quality tools: closing the gap between theory and practice. In: Proceedings of the association for information systems sigIQ Workshop, Shanghai, China, 3 December 2011
39. Zhou Y, Talburt JR (2011) Campus-Centric Distance Education using Wimba Live Classroom. In: Proceedings of the 2011 IEEE international conference on IT in medicine and education, Guangzhou, China, 9–11 December 2011
40. Pierce EM, Yonke CL, Malik P, Kagathur Nargaraj C (2012) The State of Information and Data Quality: Understanding How Organizations Manage the Quality of Their Information and Data Assets. International Association for Information and Data Quality, IAIDQ. <http://iaidq.org/publications/pierce-2012-11.shtml>
41. Lee Y, Pipino L, Funk J, Wang R (2006) Journey to data quality. MIT Publishing, Cambridge

# About the Authors

## **Laure Berti-Équille**

Institut de Recherche pour le Développement (IRD),

Montpellier, France

[Laure.Berti@ird.fr](mailto:Laure.Berti@ird.fr)

Laure Berti-Équille received an M.Sc. degree in Physics (1995) from the Toulon University, an M.Sc. degree in Computer Science (1996) from the Paris IX University and Ph.D. degree in Computer Science (1999) from the Toulon University (France). From 1999 to 2000, Laure worked as an assistant professor at the University of Avignon. In September 2000, she joined IRISA Lab as a permanent associate professor of Rennes1 University (France) and received her “Habilitation” in June 2007. From 2007 to 2009, she was a visiting researcher at AT&T Labs Research in New Jersey (USA). Her research project was supported by a Marie Curie OIF fellowship of the European Commission. Since 2011, she is a principal scientist at IRD (DR2).



## **Leopoldo Bertossi**

Carleton University,

Ottawa, Canada

[bertossi@scs.carleton.ca](mailto:bertossi@scs.carleton.ca)

Leopoldo Bertossi has been a full professor at the School of Computer Science, Carleton University (Ottawa, Canada) since 2001. Until 2001 he was a professor at the Department of Computer Science (PUC, Chile); and also the president of the Chilean Computer Science Society (SCCC) in 1996 and 1999–2000. He obtained a PhD in Mathematics from the Pontifical Catholic University of Chile (PUC) in 1988. He is a faculty fellow of the IBM Center for Advanced Studies (IBM Toronto Lab) and a member of the ACM Distinguished Speakers Program.



**Loreto Bravo**

Universidad de Concepción,  
Concepción, Chile  
[lbravo@udec.cl](mailto:lbravo@udec.cl)



Loreto Bravo is an associate professor at the Department of Computer Science, Faculty of Engineering of Universidad de Concepción, where she is also the director of the Master and Ph.D. in Computer Science. Loreto Bravo received her engineering degree in 2000 from PUC (Chile) and a Ph.D. in Computer Science from Carleton University (Canada) in 2007. She was a research fellow of the Database Group at University of Edinburgh from 2007 to 2008. She has served as PC member of international conferences such as VLDB and EDBT and reviewed articles for VLDBJ, TODS, and TKDE among others. She is the cochair of the Alberto Mendelzon Workshop 2013 and the International Conference of the Chilean Computer Science Society. Her research interests include database theory, database consistency, and logic programming.

**Reynold Cheng**

University of Hong Kong,  
Hong Kong  
[ckcheng@cs.hku.hk](mailto:ckcheng@cs.hku.hk)



Dr. Reynold Cheng is an associate professor of Computer Science in HKU. He was an assistant professor in HKU in 2008–2012. He obtained M.Sc.(CS) and Ph.D. from Purdue in 2003 and 2005, respectively. In 2005–2008, he was an assistant professor in HK PolyU. He was granted an Outstanding Young Researcher Award 2011–2012 by HKU. He received the 2010 Research Output Prize of HKU CS and the U21 Fellowship in 2011. He got the performance reward in 2006 and 2007 from HK PolyU. He is the M.Phil./Ph.D. Programme Director of HKU CS. He is an editorial member of DAPD and IS. He cochairs SSTD 2013 and has reviewed papers from SIGMOD, VLDB, ICDE, TODS, VLDBJ, TKDE and IS.

**Tamraparni Dasu**

AT&T Labs Research,  
Florham Park, NJ, USA  
[tamr@research.att.com](mailto:tamr@research.att.com)



Tamraparni Dasu is a lead member of technical staff in the Department of Statistics at AT&T Labs-Research specializing in data mining, data quality, and nonparametric statistics. She received her Ph.D. in statistics from the University of Rochester in 1991. She joined AT&T Labs-Research (Bell Labs until 1996) immediately after receiving her Ph.D. She has wide experience in mining massive

telecommunication data, data streams, and network data. Dr. Dasu is an authority on data quality and has published extensively on this topic, including the book “*Exploratory Data Mining and Data Cleaning*, T. Dasu & T. Johnson, John Wiley, NY, 2003.”

**Xin (Luna) Dong**

Google Inc.

Mountain View, CA, USA

[lunadong@google.com](mailto:lunadong@google.com)



Dr. Xin Luna Dong is a senior research scientist at Google. Prior to that, she worked as a researcher at AT&T Labs-Research. She received a Ph.D. in Computer Science and Engineering from University of Washington, and a Master's Degree in Computer Science from Peking University, China. Her research interests include databases, information retrieval and machine learning, with an emphasis on data integration, data cleaning, personal information management, and web search. She has led the Solomon project, whose goal is to detect copying between structured sources and to leverage the results in various aspects of data integration, and the Semex personal information management system, which got the Best Demo award (one of top-3) in SIGMOD'05. She has co-chaired CIKM Demo Track'13, SIGMOD/PODS PhD Symposium'12-13, SIGMOD New Researcher Symposium'12-13, QDB'12, WebDB'10, and served as a track chair for the program committee of ICDE'13, CIKM'11.

**Christian Fuerber**

Information Quality Institute GmbH,

Münsing, Germany

[cf@iqinstitute-gmbh.de](mailto:cf@iqinstitute-gmbh.de)



Christian Fürber is the founder and CEO of the Information Quality Institute GmbH, an independent consultancy for solutions related to data quality. Before founding the Information Quality Institute, Christian was a senior manager at the Data Governance Office of the German Armed Forces (GAF). Among his responsibilities was the development of GAF's data quality strategy. Since 2008, he has also been investigating the future of data quality management as an external researcher of the E-Business & Web Science Research Group. His innovations have been used in several industry projects. In line with his studies, Christian has published several papers and given talks at scientific and industrial conferences. Christian holds a master degree in Business Administration (Diplom-Kaufmann) from the Bundeswehr University in Munich (Germany).

**Lukasz Golab**

University of Waterloo,  
Waterloo, Canada  
[lgolab@uwaterloo.ca](mailto:lgolab@uwaterloo.ca)



Lukasz Golab is an assistant professor at the University of Waterloo. He joined Waterloo in 2011. Between 2006 and 2011, he was a senior member of research staff at AT&T Labs in Florham Park, NJ, USA. He obtained his B.Sc. (computer science) from the University of Toronto and his Ph.D. (computer science) from the University of Waterloo, winning the Alumni Gold Medal for top Ph.D. graduate. Lukasz's research interests are in data management and analysis, including data warehousing, data cleaning and data stream processing.

**Markus Helfert**

Dublin City University,  
Dublin, Ireland  
[markus.helfert@computing.dcu.ie](mailto:markus.helfert@computing.dcu.ie)



Markus Helfert is a lecturer in information systems at the School of Computing, Dublin City University (Ireland), Program Chair of the M.Sc. in Business Informatics and head of the Business Informatics Group. His research centres on information quality management and includes research areas such as cloud computing, IT service management, enterprise architecture, enterprise analytics and big data. Current research focuses on the evaluation of the value of information quality management in networked organizations and information quality maturity models. Markus has authored academic articles and book contributions and has presented his work at international conferences. He has served as member of program committees and chaired several international conferences. Markus holds a doctor in business administration from the University of St. Gallen in Switzerland.

**Martin Hepp**

Universität der Bundeswehr München,  
Neubiberg, Germany  
[mhepp@computer.org](mailto:mhepp@computer.org)



Martin Hepp is a professor of General Management and E-Business and head of the E-Business and Web Science Research Group at the Universität der Bundeswehr, Munich, Germany. He received his Phd from University of Würzburg, Germany in 2003, and Habilitation (venia Legendi) for Information Systems in 2008. His major research interest is in Semantics in Business Information Systems, especially the use of ontologies for advancement in the automation of business

processes. He has contributed to semantic web foundations as well as on applying semantic web technologies to address core challenges of information systems such as semantic web services and semantics-supported business process management.

**Ram Kumar**

Insurance Australia Group (IAG),

Sydney, Australia

[Ram.Kumar@iag.com.au](mailto:Ram.Kumar@iag.com.au)



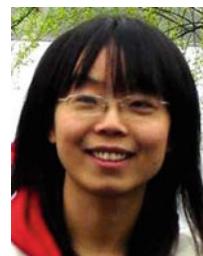
Ram Kumar is the group chief technology officer and chief information officer (Asia) of IAG. He has over 24 years of commercial experience in various fields of IT. He has consulted to various organizations in India, Australia, New Zealand, USA, China, and Thailand. Ram has contributed significantly to the development of IT Open Industry Standards through OASIS, the world's leading XML and SOA/Web Services Standards development body since 1999. He chairs an international technical committee under OASIS that develops international and open industry standards for managing Party data, and his standards are used around the world by some well-known international organizations and applications including Google Earth/Maps. Ram has published over 120 articles/papers in refereed and reputed journals and conferences.

**Pei Li**

University of Milan,

Milan, Italy

[pei.li@disco.unimib.it](mailto:pei.li@disco.unimib.it)



Pei Li is a senior researcher at University of Zurich. She received her Ph.D. education in computer science at the University of Milan-Bicocca from 2009 to 2012. She visited AT&T Labs-Research in 2010 and 2012. Previously, she obtained an M.S. (computer science) at Beijing University of Posts and Telecommunications and a B.S. (electronic engineering) at the same university. Pei's research interests are in data quality, data cleaning, and data integration, as well as large-scale data analytics and cloud computing.

**Rob Logie**

SBI General Insurance Company Limited,

Mumbai, India

[Rob.Logie@SBIGeneral.in](mailto:Rob.Logie@SBIGeneral.in)



Mr. Rob Logie commenced working full time with SBI General Insurance in India in November 2008 and was appointed to the role of deputy chief executive officer in February 2010. He has also been the IAG chief representative in India since

November 2008. With a background in accounting and business advisory in Australia and the UK, Mr. Logie has been with IAG since 1998 where he has held diverse senior roles primarily within the Direct Insurance arm. The Direct Insurance arm of IAG in Australia has gross written premium in excess of AUD 4 billion. He has over 14 years of experience in the general insurance industry.

**Andrea Maurino**

University of Milan,  
Milan, Italy  
[maurino@disco.unimib.it](mailto:maurino@disco.unimib.it)



Andrea Maurino is an assistant professor at Università di Milano Bicocca. His research interest covers many areas in the field of database systems and service science. In the field of data quality, his research interests are: record linkage, cooperative information systems, and assessment techniques of data intensive web applications. In the field of service science he focuses on the analysis of quality of services and nonfunctional properties. He is the author of more than 40 papers including international journals and conferences; he is also the author of four book chapters. He was the program cochair of QDB'09 workshop and the guest editor of IEEE Internet Computing in 2010. He is a reviewer for several journals including *Information Systems, Knowledge Data and Engineering*.

**Danette McGilvray**

Granite Falls Consulting Inc.,  
Newark, CA, USA  
[danette@gfalls.com](mailto:danette@gfalls.com)



Danette McGilvray is the president and principal of Granite Falls Consulting, Inc., a firm that helps organizations increase their success by addressing the information quality and data governance aspects of their business efforts. She is the author of *Executing Data Quality Projects: Ten Steps to Quality Data and Trusted Information*™ (Morgan Kaufmann, 2008). An internationally respected expert, her Ten Steps™ approach has been embraced as a proven method for creating and managing information and data quality. Her trademarked approach, in which she has trained Fortune 500 clients and thousands of workshop attendees, applies to all types of data and all organizations. Her book is used as a textbook in university graduate programs and is also available in Chinese.

**Ge Mouzhi**

Universität der Bundeswehr Munich  
Munich, Germany  
[mouzhi.ge@tu-dortmund.de](mailto:mouzhi.ge@tu-dortmund.de)



Mouzhi Ge is a senior researcher at the Universitaet der Bundeswehr Munich in Germany. Before joining Universitaet der Bundeswehr Munich, he had worked as an IT manager in the Oxbridge Investment Ltd. in UK and worked as a postdoctoral researcher at the Technical University Dortmund in Germany. During 2011, he is a visiting researcher at the University of Milano-Bicocca in Italy. Mouzhi Ge obtained his Ph.D. from the Dublin City University in Ireland. His research interests are mainly focused on data quality management, recommender systems and semantic web technologies. He has served as a program committee member, workshop chair or reviewer for a variety of international conferences and journals.

**Boris Otto**

University of St. Gallen,  
St Gallen, Switzerland  
[boris.otto@unisg.ch](mailto:boris.otto@unisg.ch)



Dr. Boris Otto is an assistant professor at the University of St. Gallen, Switzerland. His main areas of research are data governance, data quality management, master data management, and consumer-centric information management. His work has been published in numerous scientific journals and in proceedings of scientific conferences. Prior to his current position, Boris Otto worked for PricewaterhouseCoopers, the Fraunhofer-Institute for Industrial Engineering and SAP. Furthermore, in 2011 he was a visiting research fellow at the Tuck School of Business at Dartmouth College. Boris Otto holds a Dipl.-Ing. oec. degree from the Technical University of Hamburg-Harburg and a Dr.-Ing. degree from the University of Stuttgart. In 2012 he completed his postdoctoral qualification thesis on “Enterprise-Wide Data Quality Management in Multinational Corporations” at the University of St. Gallen.

**Elizabeth Pierce**

University of Arkansas at Little Rock,  
Little Rock, Arkansas, USA  
[expierce@ualr.edu](mailto:expierce@ualr.edu)



Elizabeth Pierce is an Associate Professor of Information Science at the University of Arkansas at Little Rock (UALR). She joined the faculty in August 2006 as part of UALR's efforts to launch the first Master of Science in Information Quality. Prior to coming to UALR, Elizabeth Pierce taught for 11 years at Indiana University of Pennsylvania. She holds a Ph.D. in statistics & management science from the

University of Michigan, a Master of Science degree in computer science from Iona College, and Bachelor of Science degrees in quantitative business analysis and mathematics from Penn State.

**Thomas Redman**

Navesink Consulting Group

NJ, USA

[tomredman@dataqualitysolutions.com](mailto:tomredman@dataqualitysolutions.com)



Dr. Thomas C. Redman, “the Data Doc,” is the president of Navesink Consulting Group. He has helped hundreds of organizations understand the importance of data and start their data programs, resulting in big sustained improvements. In doing so, they lower costs, increase revenues, improve customer satisfaction, and make more confident decisions. Tom is an internationally known lecturer and author of dozens of papers. His fourth book, *Data Driven: Profiting from Your Most Important Business Asset* (Harvard Business Press), is a *Library Journal* Best Business Book of 2008. He holds two patents. Prior to forming Navesink, Tom started and led the Data Quality Lab at Bell Labs where he and his team were first to extend quality principles to data and information.

**Heather Richards**

Canadian Institute for Health Information,

Victoria, BC, Canada

[HRichards@cihi.ca](mailto:HRichards@cihi.ca)



Heather Richards is a program consultant at the Canadian Institute for Health Information (CIHI). She works on projects related to case mix system design, health system funding models, and health system performance measurement. Heather has also managed CIHI data quality studies that aim to understand the reliability and completeness of Canada’s health information databases. Prior to CIHI she worked at Statistics Canada on coverage studies for the census of population data. Heather has previously served as the Director of Publicity for the International Association for Information and Data Quality. She has a Bachelor of Science degree in mathematics and is a certified information quality professional.

**Ken Self**

The Shell Company of Australia,

Melbourne, Australia

[Ken.Self@shell.com](mailto:Ken.Self@shell.com)



Ken Self is Shell’s Data Strategy and Standards Manager. In this role he is responsible for continually improving data processes and content standards to drive data quality improvements and enable efficient operations, effective controls and

maximum value for Shell's businesses. This includes development and implementation of data quality standards and KPIs for data processes, analyzing the cost of poor data quality, developing a data quality curriculum, and comparing Shell's data quality performance against best practices. Since joining Shell Australia in 1985, he has also spent time in Sarawak Shell and in the Shell Downstream head office in London. He is now based in Melbourne, Australia, but continues in a global role working in Shell's Data Process team whose members are located in all parts of the world.

**Divesh Srivastava**

AT&T Labs Research,  
Florham Park, NJ, USA  
[divesh@research.att.com](mailto:divesh@research.att.com)



Divesh Srivastava is the head of the Database Research Department at AT&T Labs-Research. He received his Ph.D. from the University of Wisconsin, Madison, and his B.Tech. from the Indian Institute of Technology, Bombay. He is a Fellow of the ACM, on the board of trustees of the VLDB Endowment, and an associate editor of the ACM Transactions on Database Systems. He has served as the associate editor-in-chief of the IEEE Transactions on Knowledge and Data Engineering, and the program committee cochair of many conferences, including VLDB 2007. He has presented keynote talks at several conferences, including VLDB 2010. His research interests span a variety of topics in data management.

**John Talburt**

University of Arkansas at Little Rock,  
Little Rock, Arkansas, USA  
[jrtalburt@ualr.edu](mailto:jrtalburt@ualr.edu)



Dr. John Talburt is a professor of information science and executive director of the Center for Advanced Research in Entity Resolution and Information Quality at the University of Arkansas at Little Rock where he also coordinates the information quality graduate program. He also serves as the chief scientist for Black Oak Partners, LLC, and advisor to the Board of Directors of the International Association for Information and Data Quality. He previously served as the leader for research and development and product innovation at Acxiom Corporation, a global leader in information management and customer data integration. He is the coeditor of *Data Engineering: Mining, Information and Intelligence* (Springer, 2010) and author of *Entity Resolution and Information Quality* (Morgan Kaufmann, 2011).

**Nancy White (Retired)**

Canadian Institute for Health Information,  
Victoria, BC, Canada  
[nancyellenwhite@gmail.com](mailto:nancyellenwhite@gmail.com)



Nancy White is a health information consultant based in Ottawa, Ontario. She retired in 2012 after more than a decade as a manager with the Canadian Institute of Health Information. Nancy began her career as a physiotherapist at Toronto's Hospital for Sick Children. After earning a master's degree in business administration, she led province-wide initiatives in New Brunswick including health human resource planning, health system funding, performance measurement and accountability, and data quality. In her career at CIHI, Nancy managed the development and operation of new national reporting systems in rehabilitation, home care, and facility-based continuing care.

**LwangaYonke**

International Association of Information and Data Quality  
(IAIDQ)  
Baltimore, MD, USA  
[Lwanga.yonke@iaidq.org](mailto:Lwanga.yonke@iaidq.org)



C. Lwanga Yonke is a seasoned information quality and data management leader with more than 20 years of experience. He is a member of the Society of Petroleum Engineers (SPE) and a senior member of the American Society for Quality (ASQ). Lwanga is a founding member of the International Association for Information and Data Quality (IAIDQ) and currently serves as an Advisor to the IAIDQ Board. He is the recipient of the 2011 IAIDQ Distinguished Service Award and the 2008 SPE Western North America Regional Management and Information Award. An ASQ Certified Quality Engineer, Lwanga holds an MBA from California State University and a BS degree in petroleum engineering from the University of California at Berkeley.

**Yinle Zhou**

University of Arkansas at Little Rock,  
Little Rock, Arkansas, USA  
[yxzhou@ualr.edu](mailto:yxzhou@ualr.edu)



Dr. Yinle Zhou is an affiliate member of the Graduate Faculty at University of Arkansas at Little Rock (UALR). She holds a Ph.D. in integrated computing with emphasis in information quality (IQ) from UALR where her doctoral research focused on modeling the management of entity identity information in entity resolution systems. She also holds a master of science in information quality from UALR,

a bachelor of business administration in electronic commerce from Nanjing University in China, and the Information Quality Certified Professional (IQCP) credential issued by the International Association for Information and Data Quality (IAIDQ). Her research and publications are in areas of entity and identity resolution, identity management, and information quality.

# Index

## A

- Accuracy, 1, 23, 42, 83, 87, 147–149, 165, 180, 197, 198, 217, 224, 245, 246, 264, 266, 294, 296–303, 305, 306, 309–316, 329, 330, 340, 356, 388, 404, 414  
Active integrity constraint, 192  
Algebraic model, 248  
Algorithm, 136, 137, 182, 198, 214, 216, 217, 219–223, 225, 227, 228, 231, 241–245, 248, 260–263, 275, 286–289, 293, 294, 300–301, 308–311, 313, 406  
Ambiguity, 168, 226, 272, 273  
Anomalies and outliers, 167  
Answer set program (ASP), 189–191, 193, 203–206  
Approach to data quality, 24, 55, 62, 164, 322, 323, 348, 367  
Approximate (match, matching, string matching), 147, 188, 240, 241, 246, 275  
Approximate string matching (ASM), 241  
ASP. *See* Answer set program (ASP)  
Asserted resolution, 266  
Assessment tool, 329, 330, 332, 341–343  
Attribute-based, 224, 247, 248, 260, 261, 263  
Attribute distance, 219  
Attribute uncertainty, 276

## B

- Balanced scorecard, 385, 387, 394  
Bayesian analysis, 293, 298, 302, 303, 305, 307  
Benchmark, 158, 228, 333, 336, 361, 366, 409

BICC. *See* Business Intelligence Competency

Centre (BICC)

Bootstrapping, 227, 228

Bulk update, 350, 352, 368

Business case, 13, 15, 16, 24, 25, 34, 47, 49, 56, 158, 354, 358

Business case for data quality, 13, 15, 47, 56

Business engineering, 99, 100, 110

Business impact, 8, 45, 59, 82, 86, 89, 349, 403, 408

Business Intelligence Competency Centre (BICC), 384–386, 394

Business process, 16, 22, 26, 35, 45, 63, 64, 66, 81, 83, 88, 94, 101, 103–105, 110, 111, 114, 146, 159, 349, 350, 353–355, 357, 358, 363, 365, 367, 368, 370, 375, 377, 386, 388, 389, 391, 423

Business rules, 28, 32, 45, 55, 59, 66, 79, 80, 103, 137, 351, 357, 359, 362, 363, 402

Business value of data quality, 83–85

## C

Career paths, 403–405

Categorical value, 152, 297

Cautions, 71

CDO. *See* Chief data office (CDO)

CDs. *See* Conditional dependencies (CDs)

Certain answer, 192–194, 202

Certification, 8, 52, 367, 401, 407–409, 414

CFD. *See* Conditional functional dependency (CFD)

Chase, 194, 202–205

Chief data office (CDO), 35, 410

Choice node, 225

- CIND. *See* Conditional inclusion dependency (CIND)
- Clean answer, 202, 203, 205
- Cleaning, 87, 133, 136, 137, 139, 164–166, 169, 171, 173–176, 181–183, 186, 188, 189, 197–198, 203, 205–208, 214, 224, 240, 250, 252–256, 272, 274–279, 282–289, 312, 362, 398, 421–423
- Cleaning algorithms, 272, 276
- Cleaning methods, 173–175, 177, 276
- Clinical data, 323, 324, 329, 345
- Cluster, 6, 213, 214, 216, 219, 225–228, 231, 247, 248, 252, 260–266, 336
- Clustering, 214, 219, 227, 228, 230–232, 248, 266, 339
- Co-copying, 302, 303, 312
- Communication, 5, 26, 27, 31, 36, 45, 56, 63, 69, 72, 82, 95, 137, 138, 143, 349, 361, 380, 404, 415, 421
- Communities of Practice, 409, 413, 415
- Company A, 54–61, 69, 70, 72
- Company B, 61–70, 73
- Complex data glitches, 167–169
- Compound keyword, 226, 227
- Comprehensive and Consistent Customer Views, 392
- Concepts, 5–7, 13, 44–47, 49, 50, 53, 64, 67, 94–96, 98, 105, 143, 157, 158, 165, 248, 398–400, 405, 406
- Concomitant glitches, 169
- Conditional dependencies (CDs), 195–198, 201
- Conditional functional dependency (CFD), 123, 127, 195–197
- Conditional inclusion dependency (CIND), 196, 197
- Conferences, 4, 43, 100, 221–223, 328, 361, 406, 409, 413–415, 420–425, 427
- Confidence of values, 299, 309, 310
- Configuration, 99, 147, 258, 266
- Conflict resolution, 383
- Connection strength, 225, 226
- Consistent database, 148
- Consistent query answering (CQA), 186–188
- Context information, 214, 217, 221
- Continuous improvement, 19, 85, 265, 323, 326, 327, 329, 333–334, 336, 342, 349, 351, 353, 358, 360, 361, 364, 367, 368
- Continuous improvement strategies, 329, 333–334
- Cookbook, 55, 69
- COPDQ. *See* Cost of poor data quality (COPDQ)
- Copier, 295, 296, 302, 303, 305, 308, 310, 311, 314, 316
- Copy detection, 295, 302–306, 311–312
- Co-reference resolution, 237
- Corporate asset, 63, 373, 383
- Corporate balanced scorecard, 385, 394
- Correlated copying
- Cost and value identification, 80–82
- Cost of poor data quality (COPDQ), 24, 351, 362
- CQA. *See* Consistent query answering (CQA)
- Critical success factors, 63, 72
- Culture, 1, 8, 10, 17, 20, 21, 30, 33, 35, 36, 44, 45, 61, 62, 66, 70, 72, 89, 360, 369–395, 403, 408, 415
- Customer centricity, 394
- Customer relationship management, 236, 237
- D**
- DAMA. *See* Data Management Association (DAMA)
- Dashboard, 55, 58, 59, 70, 106, 110, 126, 127, 130, 133, 388
- Data architecture management, 99
- Data assets, 15, 95, 372, 380, 415
- Database hardening, 237
- Database maintenance, 185, 186
- Database repair, 188–191, 200, 207
- Data cleaning, 87, 133, 136, 139, 164–166, 171, 173, 175–177, 181–183, 188, 189, 197–198, 207, 208, 214, 224, 250, 252–256, 274–276, 279, 282, 284, 286–289, 398, 421–423
- Data-cleaning strategies, 175–177
- Data cleanse(ing), 86, 95, 146, 238, 257, 356, 362, 397–400
- Data collection, 19, 24, 38, 167, 322, 323, 334–336, 338–339, 343–345
- Data completeness, 124, 127, 129–131, 388
- Data creators, 20, 21, 23, 26, 27, 414
- Data currency, 124, 125, 128–129, 138, 331
- Data customers, 23, 36, 350, 353, 365
- Data defined, 22–23
- Data dependencies, 194–198
- Data editing, 182, 207
- Data freshness, 1, 124, 125, 127–128, 132, 138
- Data fusion, 293–317
- Data governance, 45, 54, 55, 57–60, 62, 69, 93–115, 392, 400, 403, 405, 414
- Data governance arrangement, 111, 114
- Data governance council, 57–59, 400

- Data governance office (DGO), 55–57  
Data imprecision, 272  
Data integration, 8, 10, 44, 123, 124, 168, 186, 191–194, 198, 213, 236, 237, 316, 317, 345–346, 370, 402, 421, 423, 427  
Data lifecycle management, 95, 99  
Data management, 2, 8, 37, 53–57, 61, 63, 66, 95–96, 98, 99, 101–105, 107, 108, 121, 158, 183, 208, 236, 247, 252, 253, 267, 348, 349, 352, 354, 356–358, 361, 363, 366, 368, 370, 372, 382, 392, 400, 412–414, 422, 427  
Data Management Association (DAMA), 53, 95, 113, 413  
Data Management Book of Knowledge (DMBOK), 53  
Data managers, 9, 357, 358, 363  
Data markets, 16, 24, 30, 34  
Data model, 22, 23, 26, 27, 29, 32, 56, 148, 183, 272, 276, 277, 279, 346, 381, 389, 393, 399  
Data monitoring, 55–60, 68, 70, 104–106, 109, 112, 114, 126, 129–132, 134, 149–153, 155, 271, 272, 274, 283, 284, 322, 328–343, 361, 366, 376, 383, 387, 392, 402  
Data precision, 23, 25, 124, 155, 157, 226, 227, 273, 312–314  
Data profiling, 238, 239, 253, 255, 256  
Data quality assessment, 5, 10, 76, 80, 150–153, 164, 165, 172, 181, 182, 208, 250, 252–256, 329, 398  
Data quality constraint, 127, 135, 149  
Data quality cost, 10, 77–82, 85–89, 375  
Data quality cost and value indicators, 87, 88  
Data quality cost and value management, 75–89  
Data quality cost and value model, 85–87  
Data quality cost detection, 78, 86, 87  
Data quality cost prevention, 78, 81, 85, 87, 322, 326, 327, 333, 335, 371  
Data quality dashboard, 55, 58, 59, 70, 106, 110, 126, 127, 130, 133, 388  
Data quality definition, 18–19, 145  
Data quality dimensions, 7, 45, 83, 87, 95, 147, 148  
Data quality education, 406, 409–414  
Data quality framework, 56, 76, 329, 330, 332, 392  
Data quality improvement, 17, 78, 88, 263, 275, 284–289, 333, 368, 383, 402, 403, 415  
Data quality management (DQM), 1–11, 15–39, 54, 75–77, 81, 82, 85–89, 95–97, 99, 108, 113, 114, 141–159, 181–208, 263, 357, 367, 380, 382, 387, 398, 399, 406, 409, 411, 421, 422, 425  
Data quality metrics, 2, 5, 56, 59, 60, 133  
Data quality problems, 8, 43, 44, 47, 60, 61, 76–80, 86, 88, 123, 125, 126, 128, 132–137  
Data quality process, 3, 108, 125, 165, 166, 356, 177, 337  
Data quality professional, 401–414  
Data quality program, 15, 20, 21, 24, 26, 29, 30, 41–43, 47–59, 61–65, 68–71, 87, 322, 326, 352, 364, 367, 368, 400, 415  
Data quality program framework, 49–52, 64–65  
Data quality project(s), 41–72, 86, 87, 89, 402, 424  
Data quality repair costs, 78, 86  
Data quality services (DQS), 51, 55–57, 59, 60, 272  
Data quality skills, 52–53, 415  
Data quality standards (DQS), 3, 78, 350, 351, 354, 356, 363, 380  
Data quality team, 42, 48, 53, 63, 66, 67, 72  
Data quality tool (DQT), 49, 56, 57, 63, 68, 70, 148, 352, 359, 363, 392, 411, 415  
Data quality value, 82–85, 88  
Data redundancy, 235  
Data repair, 86, 164, 170, 173–175  
Data requirement(s), 20, 144–154, 159, 350, 353  
Data requirement types, 147, 148, 150  
Data sharing, 27, 36  
Data source, 2, 81, 100, 105, 108, 122, 125, 129, 130, 133, 136, 139, 142, 146, 186, 192–194, 237, 253–300, 302, 303, 305–310, 315, 316, 376, 399  
Data stakeholders, 145, 400  
Data staleness, 127  
Data standards, 26, 27, 250, 254, 323, 326–328, 332, 334, 335, 337, 345, 400  
Data stewardship, 108, 400, 405  
Data strategy, 99, 108, 426  
Data stream, 122, 169–171, 421, 422  
Data valuation, 34  
Data values, 22, 23, 28, 31, 167, 174, 176, 182, 196, 316

- Data warehouse, 10, 32, 43, 44, 60–63, 65, 67, 71, 81, 121–139, 398
- Decision-making, 16, 18, 25, 34, 35, 38, 61, 76, 77, 81, 83, 84, 88, 96, 145, 198, 231, 328, 370, 371, 383, 393, 400, 403
- Decision-making rights, 93–95
- Decision quality, 83, 84
- De-duplication, 174, 236
- Delphi algorithm, 216
- Dependency graph, 222, 223
- Deterministic matching, 240
- DQM-Vocabulary, 150–152, 154, 155
- DGO. *See* Data governance office (DGO)
- Dimensional hierarchy, 216
- Dimensions of Data Quality, 18, 23–24
- Direct copying, 302, 312
- Disagreement decay, 230, 231
- Disambiguation, 224, 227, 236
- Distributed data quality, 125, 127–128
- DMAIC cycle, 20
- DMBOK Functional Framework, 53
- DMBOK. *See* Data Management Book of Knowledge (DMBOK)
- Domination, 199, 201, 202, 206
- DQM-Vocabulary, 150–152, 154, 155
- DQS. *See* Data quality services (DQS); Data quality standards (DQS)
- DQT. *See* Data quality tool (DQT)
- Drivers and Alignment, 50, 51
- Duplicate detection, 8, 10, 170, 174, 207, 216, 237, 242
- Duplicates, 147, 166–168, 170, 171, 174, 175, 198, 207, 215–217, 219, 277
- Dynamic programming, 275, 287
- E**
- Early binding, 231
- EDD. *See* Enterprise Data Dictionary (EDD)
- Edit distance, 197, 258, 301
- EIS. *See* Entity identity structure (EIS)
- Electronic health record, 322, 323, 334, 344, 345
- EMD. *See* Enterprise master data (EMD)
- Enterprise architecture, 382, 422
- Enterprise Data Dictionary (EDD), 56, 389–391, 393
- Enterprise data management strategy, 57
- Enterprise data model, 56, 389
- Enterprise information management, 370, 373, 377, 382, 394, 395
- Enterprise master data (EMD), 103–105, 107–110
- Entity-based query, 274
- Entity identity information management (EIIM), 246–248, 263–267
- Entity identity integrity, 236, 246, 265
- Entity identity structure (EIS), 238, 247–249, 261, 266, 267
- Entity reference, 237–241, 245
- Entity relationship analysis, 248
- Entity relationship graph, 248
- Entity resolution (ER), 7, 8, 10, 182, 183, 198–207, 236–240, 245–249, 251, 252, 258, 260–265
- Entity resolution system, 248
- Entrepreneur, 42
- Equivalence errors, 214, 215
- Error detection, 10, 133, 136, 139
- ER. *See* Entity resolution (ER)
- Extract–Transform Load (ETL), 65, 122, 123, 125, 128, 167, 174, 239
- F**
- Fact table, 122–124, 128, 131, 134
- FBS. *See* Feature-based similarity (FBS)
- FD. *See* Functional dependency (FD)
- Feature-based similarity (FBS), 214, 224
- Financial and statistical data quality, 340–343
- FIQ. *See* Framework for Information Quality (FIQ)
- Fitness for use, 3, 18, 95, 329, 332
- Foreign key constraint, 184
- Foundation, 9, 13, 32, 43, 44, 49–51, 58, 60, 65–71, 86, 94, 146, 236, 324, 335, 340, 388, 408
- Framework for Information Quality (FIQ), 45–47
- Functional dependency (FD), 133, 136–138, 184, 185, 187–190, 194, 195
- Fuzzy matching, 240
- G**
- Glitch detection, 168–171, 173
- Glitch signature, 166, 172
- Global-positioning system (GPS), 33, 179, 271
- Glue records, 262
- Governance, 8, 10, 14, 35, 36, 45, 54, 55, 60, 62, 69, 70, 93–115, 247, 354, 355, 378–387, 392, 394, 400, 403, 405, 408, 414, 421, 424, 425
- GPS. *See* Global-positioning system (GPS)
- Greedy algorithm, 275, 289
- Group similarity, 219
- Guideline for cost and value analysis, 87–88

**H**

- Habits of data quality, 20, 21  
HAC. *See* Hierarchical agglomerative clustering (HAC)  
Health system data, 322, 323, 325, 326, 334, 345  
Heuristics, 128, 135, 170, 174, 222, 275, 276, 283, 288–289  
Hidden Markov Model (HMM), 312  
Hierarchical agglomerative clustering (HAC), 228  
HMM. *See* Hidden Markov Model (HMM)

**I**

- IAG. *See* Insurance australia group (IAG)  
IAIDQ. *See* International Association for Data and Information Quality (IAIDQ)  
IC, 184, 186, 191, 193  
Identity attributes, 245, 263  
Identity capture, 266  
Identity resolution, 237, 429  
Identity rules, 253, 256, 261, 262  
Identity uncertainty, 237  
Identity update, 266  
IMCC. *See* Information management competency centre (IMCC)  
IMCF. *See* Information management custodians forum (IMCF)  
IMF. *See* Information management framework (IMF)  
IMG. *See* Information management group (IMG)  
Inclusion dependency, 184, 195, 196  
Incomplete data, 79, 131, 132, 194  
Inconsistent and Faulty Data, 167  
Inconsistent database, 186–189, 277  
Independency graph, 222  
Independent source, 297, 302, 303, 305, 307, 310, 330  
Indirect copying, 303  
Inferencing notation, 149  
Information and Data Quality (IAIDQ), 8, 52, 367, 401–409, 412–414, 428, 429  
Information architecture, 8, 397, 399, 402, 408  
Information asset, 370–373, 377–381, 383, 386, 394  
Information centric organisation, 369–395  
Information extraction, 238, 239  
Information life cycle, 45, 68, 267, 373, 375, 379, 381, 383  
Information management awareness, 374, 378–379 culture, 371, 373, 377, 378, 380, 387  
governance, 379, 383, 385, 387, 394  
governance framework, 383–385  
strategy, 370, 373, 377–380, 382, 383, 386, 392, 393  
Information management competency centre (IMCC), 383–386  
Information management custodians forum (IMCF), 385  
Information management framework (IMF), 379, 381–383  
Information management group (IMG), 384, 385, 387–390, 393  
Information product, 81, 88, 397, 399  
Information quality environment, 403, 408  
monitoring, 387  
professional, 42, 44, 52, 426  
strategy, 8, 403, 408, 421  
value, 8, 403, 408  
Information Quality Certified Professional (IQCP), 52, 367, 401, 406–409, 413  
Information quality management (IQM), 380, 382, 387, 388, 399, 422  
Information theory, 31–33, 281  
Insurance Australia Group (IAG), 374, 375, 423, 424  
Integrity constraint, 2, 9, 123, 124, 182–185, 191, 277  
Integrity constraint enforcement, 191  
International Association for Data and Information Quality (IAIDQ), 8, 52, 367, 401–409, 412–414, 428, 429  
Inter-object relationship, 223, 224, 232  
Intrapreneur, 42  
IQCP. *See* Information Quality Certified Professional (IQCP)  
IQM. *See* Information quality management (IQM)  
IS/IT business value, 83  
Issue tracking, 70  
Iterative record linkage, 217–219

**J**

- Jaccard coefficient, 241–242  
Jaro distance, 243

**K**

- Key constraint, 184, 185  
Key performance indicators, 387

**L**

- Late binding, 231–232  
Leadership, 20, 21, 26, 35, 71, 322, 371, 415

Lean Sigma, 351, 353, 361–364, 367  
 Levenshtein distance, 242  
 Life cycle, 42, 44, 45, 68, 69, 82, 94, 95, 97, 100, 105, 108, 114, 265, 267, 350, 353, 355, 373, 375, 379, 381, 389, 402, 403  
 Lineage, 2, 7–10, 179, 276, 294, 315–317, 373  
 Linkage, 2, 5, 8–10, 157, 213–215, 217–223, 228–232, 236, 272, 317, 331, 424  
 Linked object, 217, 223  
 Linked open data, 156

## M

Management system for data, 15–17, 20–21, 31, 34–38  
 Managing metadata, 366  
 Massachusetts Institute of Technology (MIT), 24, 224–226, 295, 409–410  
 Master data, 9, 43, 44, 99, 102–108, 110, 111, 179, 207, 236, 247, 252, 267, 349, 350, 352–358, 360–363, 365, 366, 368, 372, 391, 392, 400, 414, 425  
 Master data management (MDM), 99, 100, 236, 247, 252, 267, 349, 352, 354, 356–358, 361, 363, 366, 368, 391–393, 400, 414  
 Master reference data (MRD), 349–361, 365, 367, 392  
 Matching dependencies (MDs), 183, 200, 201, 204–207, 324  
 Matching function, 182, 200, 201, 203, 205  
 Materialized views, 121, 122, 124, 125, 127, 128, 131, 136, 138, 139  
 Material master dashboard, 106  
 Maturity assessment, 354, 361  
 Maturity model, 14, 86, 112–114, 354, 358, 361, 400  
 Maturity model for data governance, 112–114  
 MDM. *See* Master data management (MDM)  
 MDs. *See* Matching dependencies (MDs)  
 Measurement, 3, 8, 28, 29, 32, 77, 83, 88, 124, 131, 135, 139, 145, 146, 166, 167, 170, 264, 329, 330, 345, 356, 358, 359, 362, 363, 367, 387, 402, 408, 415, 426, 428  
 Measurement error, 329, 330  
 Mediator, 191, 192, 198  
 Merge closure, 199, 206  
 Merging, 81, 123, 182, 199, 200, 204, 205, 207, 219–221, 237, 260, 277, 316  
 Metadata, 23, 29, 36, 38, 45, 55–57, 66, 152, 194, 316, 332, 362, 366, 370, 380, 402

Meta-object facility (MOF), 113  
 Misfielding, 147, 156, 259  
 Missing data, 79, 129–131, 167  
 MIT. *See* Massachusetts Institute of Technology (MIT)  
 MOF. *See* Meta-object facility (MOF)  
 Monetizing data, 31, 33–34, 36  
 Monitoring data quality, 57, 109, 153, 337–338  
 MRD. *See* Master reference data (MRD)  
 Multi-occurred glitches, 169  
 Multi-type glitch, 169

## N

Null value, 23, 188, 258

## O

Ontology(ies), 10, 141–143, 149–151, 154–159, 194, 207, 208, 422  
 OWL. *See* Web ontology language (OWL)

## P

Patterns, 86, 130, 133–136, 165, 166, 168–172, 174, 177, 195–197, 228, 238, 239, 241, 251, 253, 255–257, 376, 388  
 Practitioner(s), 8, 11, 19, 21, 25, 77, 83, 87, 93, 95, 100, 115, 352, 353, 358, 363–367, 387, 398, 399, 406–410, 413–415  
 Process improvement(s), 16, 20, 81, 112, 353  
 Program (data quality). *See* Data quality program  
 Project (data quality). *See* Data quality project  
 Provenance, 2, 9, 10, 136, 143, 149, 157, 159, 248, 315

## R

RDF. *See* Resource description framework (RDF)  
 Record linkage, 2, 8–10, 213–215, 217–223, 228–232, 236, 272, 317, 424  
 Reference data, 45, 148, 155, 156, 159, 349, 392, 393, 402  
 Representation(al) consistency, 23, 147  
 Resource description framework (RDF), 142, 143, 149, 150, 152–154, 156–159, 207

## S

Sampling, 127, 176, 338, 356, 357, 359  
 Semantic, 2, 23, 80, 119, 129, 141–159, 181, 240, 274, 315, 376, 400, 422

- Semantic constraint, 182, 183, 186, 200  
Semantic definition of data, 119, 143, 148, 154–155  
Semantic network(s), 142, 157  
Semantic Web, 119, 141–159, 207, 315, 423  
Semantic Wiki, 152  
Sensor(s), 2, 124, 130, 133, 136, 179, 271, 272, 274, 277, 283, 284, 370, 376  
Sensor-monitoring system, 272  
Sequence neutral, 260  
Service(s), 8, 15, 16, 27, 28, 33, 39, 43, 50, 51, 54–65, 67–71, 76, 85, 94, 119, 123, 170, 272, 315, 322–326, 336, 337, 340, 341, 345, 349–351, 353, 357, 358, 360, 365, 367, 374–376, 388, 389, 393, 398, 404, 411, 422–424, 428  
Service centre, 349, 353, 357, 358, 360, 365, 367  
Service oriented architecture, 119, 388, 389  
Service quality, 51, 55–57, 59, 60, 272  
Shell, 319, 347–368, 426, 427  
Similarity analysis, 245–246  
Similarity decisions, 222  
Similarity function, 215, 216, 240, 241, 244–245, 257, 258  
Single source of truth, 372, 391–393  
Six sigma, 20, 24, 107, 109, 398  
Skyline queries, 289  
Smith-Waterman distance, 242–243  
Social, 2, 15, 16, 24–26, 36, 38, 42, 121, 124, 139, 157, 249, 252, 253, 370, 376, 401, 409, 415  
Social impediments, 25–28  
Social media, 42, 121, 124, 129, 157, 376, 401, 415  
Social network(s), 2, 38, 370  
Source accuracy, 296, 300, 306, 309, 310, 313–316  
Source dependence, 308, 309  
SPARQL Inferencing Notation (SPIN), 149, 150, 152, 153  
SPARQL Protocol and RDF Query Language (SPARQL), 149–154, 156  
SQL. *See* Structured query language (SQL)  
Stakeholder, 42, 65, 66, 68, 70, 84, 143, 145, 321, 322, 326, 327, 332–334, 336, 340, 344, 345, 351, 361, 363, 372, 373, 380, 400, 403  
Standardize, 149, 153, 159, 256, 257, 324–326, 337, 340, 398, 400, 406  
Stanford Entity Resolution Framework (SERF), 248  
State Bank of India, 319, 369–395, 423  
Statistical data quality process, 165  
Statistical distortion, 166, 171, 176, 177  
Statistical process control, 80, 411  
Strategy, 8, 18, 49, 77, 99, 128, 165, 226, 245, 300, 326, 370, 400, 421  
Stream data warehouse, 122, 123  
Strong-boolean-valued neighbor, 222, 223  
Structured query language (SQL), 70, 121, 127, 149, 150, 182, 240, 253  
Success of information systems, 77  
Survivor record EIS, 247  
Swoosh, 182, 199–200, 206–207, 263  
Syntactic, 147, 149, 186, 188, 194, 205, 240, 241, 244, 254  
System edit, 327
- T**  
Talburt-Wang Index (TWI), 264, 265  
Taxonomy, 77–80, 89, 94, 315, 408  
Taxonomy for data quality costs, 78  
Technical standards, 326, 327, 342  
Technology, 1, 2, 5, 8–10, 16, 25, 26, 33, 35, 38, 52, 53, 56, 62, 63, 65, 66, 68, 71, 89, 96, 112, 119, 141–159, 238, 247, 294, 319, 325, 326, 347, 352, 356, 359, 363, 366, 370–373, 376, 377, 379–381, 384, 385, 388–393, 395, 405, 406, 409–412, 423, 425, 427  
Temporal clustering, 230  
Temporal consistency, 124, 131–132  
Temporal record linkage, 228–232  
Ten Steps<sup>TM</sup>, 44, 49, 55, 56, 424  
Ten Steps to quality data and trusted Information<sup>TM</sup>, 13, 44, 56, 424  
The Ten Steps<sup>TM</sup> methodology, 44–48, 56  
Theory, 28, 31–33, 42, 85, 96, 98, 110, 112, 165, 169, 201, 245, 248, 281, 377, 406, 409, 411, 420  
Timeliness, 13, 23, 95, 147, 149, 329, 331, 333, 358  
Tool selection, 68  
Top-*k* queries, 276  
Total quality management (TQM), 81, 85, 97, 263, 398, 399, 409, 411  
Training, 25, 33, 35, 37, 43, 49, 53, 56, 57, 60, 66–70, 78, 81, 95, 114, 227, 239, 328, 333, 334, 336–338, 340, 342, 344, 353, 357, 358, 360, 363–368, 379, 380, 401, 403, 405–407, 409–414  
Transitive closure, 260, 262, 264  
Transitive copying, 302, 303, 312  
Truth discovery, 295, 296, 302, 313, 316, 317

Trusted Reference Data, 155–156  
Trustworthiness, 124, 294–296, 301–302, 315,  
    316  
TWI. *See* Talburt-Wang Index (TWI)

**U**

UALR. *See* University of Arkansas at Little Rock (UALR)  
Uncertain databases, 272, 273, 276  
Undocumented Data, 167–168  
Uniform false-value, 297, 304  
Uniqueness, 30, 149, 167, 250, 251, 256, 351  
Uniqueness of data, 30, 149  
University of Arkansas at Little Rock (UALR),  
    409–412, 425, 428  
Unstructured data, 32, 157  
Unstructured information, 143, 157–159, 237

**V**

Value confidence, 272, 300, 302, 311

Value of high-quality data, 366  
Value similarity, 229, 301  
VDIS. *See* Virtual data integration systems  
    (VDIS)  
View maintenance, 127, 130, 186  
Violation view, 186  
Virtual data integration, 191–194, 198  
Virtual data integration systems (VDIS), 198  
Vote count, 299, 306–309

**W**

Weak-boolean-valued neighbor, 222, 223  
Web ontology language (OWL), 142, 143, 207  
Wiki Trusted Reference, 155–157, 159  
Workflow, 33, 104–108, 110, 111, 315, 352,  
    356, 358, 359, 368

**X**

X-form, 275, 281–283, 285