Lecture Noteson Business Intelligence

Fundamentals of Information Systems (University of Nairobi)

Lecture Notes on Business Intelligence

2016

Benoît Depaire

Lecture Notes on Business Intelligence

Benoît Depaire

This book is for sale at http://leanpub.com/LectureNotesonBusinessIntelligence

This version was published on 2016-12-22

Contents

An Introduction to Business Intelligence **1**
- Origins and Foundations of Business Intelligence 1
- A Business Intelligence Framework 4
- The future of Business Intelligence 9
- Resources 11

Business Intelligence Architecture **13**
- Introduction to Software Architecture 13
- BI Architecture Basics 15
- Data Warehouse Architecture 21
- Resources 23

Data Warehouse Architecture **24**
- Dimensional Modeling Fundamentals 24
- Fact Tables 25
- Dimension Tables 28
- Slowly changing dimensions 30
- ETL Subsystems 33
- Resources 35

Business Intelligence, Business Strategy and Performance Management **36**
- Recap 36
- Critical Success Factors 38
- Balanced Score Card and Strategy Maps 40
- Performance Prism 40
- Key Performance Indicator Modeling 40
- Resources 40

An Introduction to Business Intelligence

Understanding BI is the first step in reaping its potential, which is enormous by the way.

Origins and Foundations of Business Intelligence

You can't contemplate the potential of BI without first understanding what BI is. Unfortunately, there is no clear definition

Why do we need Business Intelligence?

- Making decisions are inherent to doing business
  - Structured versus unstructured decisions
  - Routine versus unique decisions
  - Decisions at operational, managerial and executive level
- Making bad decisions might have a substantial impact on a firm's performance, depending on the frequency and organizational level of decision
  - Operational routine decisions, such as "deciding which invoices to pay today", might have a low individual impact, such as receiving a discount when paid within 10 working days, but due to the routine nature of these decisions, the total impact might become substantial.
  - Strategic decisions such as entering a new market or not, typically requires large amounts of investments and could have a big impact when the decision is wrong.
- Ideally, these decisions are based on facts and knowledge and not purely on intuition.
- But what if a manager doesn't realize he should make a decision because he is unaware of a potential problem/opportunity which requires his attention?
  - Managers often have a fragmentary and biased knowledge of their business processes
  - Some insights are obscured by nature (e.g. competitor-related knowledge)
  - There is simply too much data to spot the issues or opportunities
- So the question is: **How can we use IT to improve decision making?**

History and foundations

- Database technology

  - 1969: Relational Model introduced by Codd, gave birth to our modern relational database systems.
  - For the first time, users didn’t need to know how the data was internally stored on the machine, which allowed for an explosion of the use of databases
  - Two men saw the opportunities and build the first usable relational database systems: Michael Stonebraker (Ingres/Postgres) and Larry Ellison (Oracle)
- Rise of Enterprise Systems
  - 70s: Material Requirement Planning (MRP) system
    - Production Planning
    - Purchase Planning
  - 80s: Manufacturing Resource Planning (MRP II) system
    - Extended MRP with additional focus on other aspects of manufacturing (capacity planning, standard costing, lot traceability, ...)
  - 90s: Enterprise Resource Planning
    - A suite of integrated applications (modules) to support many different business functions
      - Financial accounting (general ledger, cash management,...)
      - Management accounting (budgeting, ABC, ...)
      - Human resourcing (recruiting, payroll, ...)
      - Manufacturing (scheduling, capacity, ...)
      - Order Processing (pricing, inventory, shipping, ...)
      - Supply Chain Management (purchasing, warehousing, ...)
      - Customer Relationship Management (sales, marketing, ...)
      - ...
  - 90s: CRM systems and SCM systems
    - While ERP systems contain also these modules, these enterprise systems are dedicated to specific aspects, such as the customer and the supply chain.
  - Important to realize that databases are key in the history of EIS. All these EIS rely on a central database which integrates information from different corners of the enterprise.
- Operations Research
  - Discipline that deals with the application of advanced analytical methods to help make better decisions.
  - Originated in the military during World War I and became even more important during World War II
  - Found its way to the business domain after WW II and became the foundation for the field of Management Science
  - Mainly focused on optimization
- Artificial Intelligence
  - AI = Emulation of human intelligence
  - Field was founded in 1956 at a conference at Dartmouth College
  - AI contains different subfields, such as:
    - Problem Solving

        - Automated Reasoning
        - Knowledge Representation
        - Planning
        - Machine Learning (inducing knowledge from past experiences)
    - In the mid 60s, the experts in AI predicted that within 20 years machines would be capable of doing any work a man can do. This prediction was too optimistic and turned out to be false.
    - In the 80s the field of AI produced commercial systems called *expert systems*, which tried to simulate the knowledge and skills of a human expert. While promising at first, the technology (and AI again) failed to deliver on its promise and fell into disrepute.
    - In recent years, AI has received again a lot of attention, in particular due to the subfield called "Deep Learning". Deep Learning exploits recent increases in computing power and data volume. Some public successes of Deep Learning are:
        - Winning the Jeopardy! quiz show by IBM's Watson against the two greatest Jeopardy champions.
        - Winning the game Go against Go champion Lee Sedol by AlphaGo.
    - With the renewed interest in AI, the claims that machines will be capable to take over human work are starting to reappear again.
- Data Mining is the use of computer algorithms to extract useful knowledge and new insights from large amounts of data.
    - This field was born in the late 90s and were inspired by the fields of Machine Learning and Statistics, but with a different goal in mind.
    - Data Mining has found its way into the business environment to extract useful knowledge from the data a company possesses.
- Decision Support Systems
    - Decision support system is a general term for any computer application that enhances a person or groupâ€™s ability to make decisions.
    - The history of DSS dates back to mid 60s
    - There are different types of DSS, among which
        - Model-driven DSS
            - Based on a mathematical model (financial, optimization, simulation)
            - Spreadsheet DSS are often model-driven
            - Many of the early DSS were model-driven
        - Data-driven DSS
            - Accesses and manipulates historical data (mostly internal data)
            - Business Intelligence are actually data-driven DSS!
        - Knowledge-driven DSS
            - Contain expertise, which consists of knowledge about a particular domain, understanding of problems within this domain and some skill to solve some of these problems
            - Suggest or recommend actions to managers
            - Expert Systems are Knowledge-driven DSS

Business Intelligence and Business Analytics

- There are many different definitions on Business Intelligence, but for the purpose of this course we will define it as follows:
  - Business Intelligence is a system which relies on people, processes and technology to gather, explore, analyze, visualize and interpret business data to help an enterprise better understand its business and market and supports decision making at varying organizational levels in order to create desired business value.
- Similar to BI, Business Analytics also goes by different definitions, mainly because it is still a rather recent concept. For this course, we will define Business Analytics as follows:
  - Business Analytics is the process of applying BI, typically to a specific business function/process, to create actionable knowledge and identify new business opportunities.
  - The term BA is often renamed according to the specific function/process to which it is applied: e.g. marketing analytics, promotion analytics, customer attrition analytics)
  - Do not confuse Business Analytics with Business Analysis. The latter refers to the act of analyzing the functions and processes within an organization and focusses on process architecture and enterprise architecture.

A Business Intelligence Framework

Technology is the foundational element of Business Intelligence, but Business Intelligence is not about technology

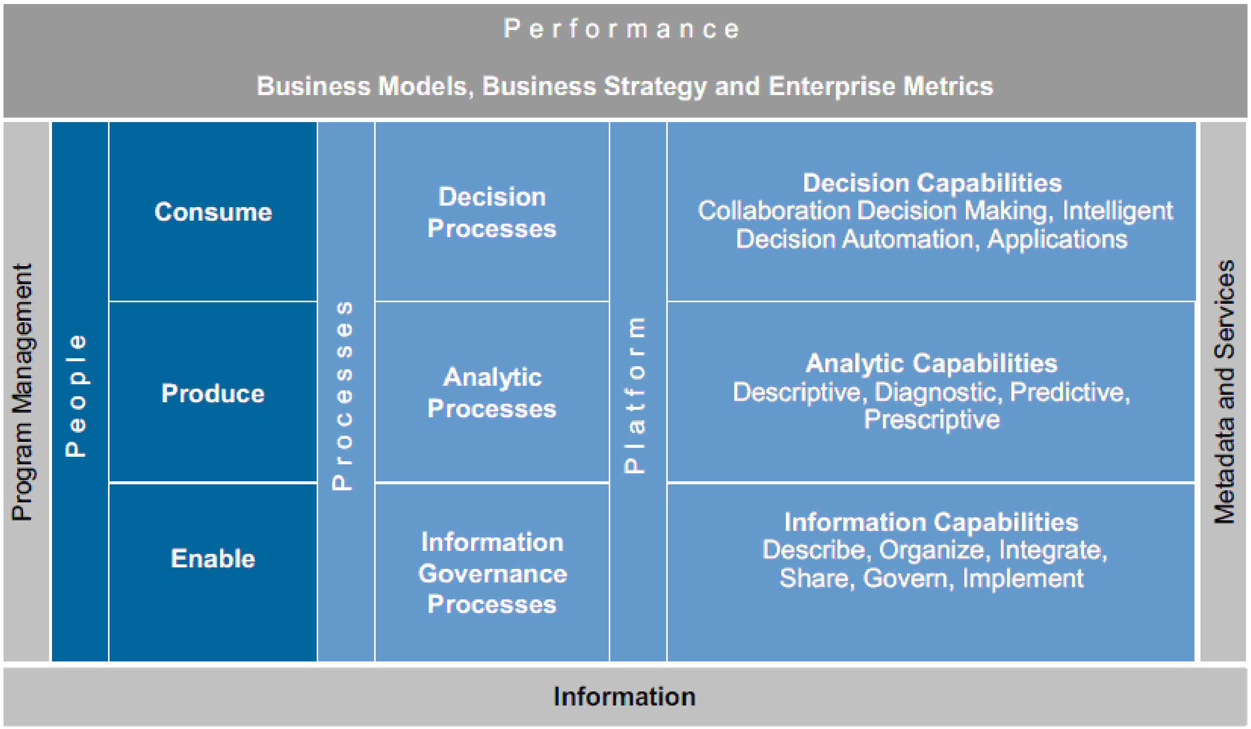
Why BI should be strategic, not tactical

- Originally, BI was used as a tactical tool, i.e. BI initiatives were responses to an individual manager's problem and need for insights from data.
- BI as a tactical solution is suboptimal because:
  - It results in separate BI initiatives which might fail to be interconnected, use incompatible technologies and use different interpretations and measurements of data concepts (e.g. lifetime value of a customer)
  - Tactical solutions might optimize local decision making, but provides no guarantee that it contributes to the company-wide strategy or that the decisions made are also optimal at a company-wide level
  - Governance of information and decision-making processes becomes very hard when BI-initiatives are scattered across the organization.
    - Recent regulations, such as Sarbanes-Oxley act, require that governance of information and decision making is on par with governance of transaction processing.
    - When decisions are made based on insights from data, it is crucial to be able to track how the insights were generated from which data.

- As IT has been infiltrating all levels of the organizations, the need for technology and data to support decision making is increasing, which will require prioritization of BI projects. Such prioritization requires a strategic approach.
  - Based on the cost, need and to which extent the BI project positively affects the strategic business objectives, priorities will be set.
  - BI as a strategic asset transforms an organization because it acts as a driver of processes, technology and service decisions.

A BI Framework

- If a company wants to use Business Intelligence strategically, it requires a framework to fully understand the different elements of strategic BI.
- For this course, we will use Gartner's Business Analytics Framework, which identifies three main components which should be aligned and integrated:
  - People
  - Processes
  - Platforms
- Note that according to the definitions we use, Business Analytics Framework is a misnomer and should rather be called Business Intelligence Framework.


Performance
Business Models, Business Strategy and Enterprise Metrics
Program Management
People
Consume
Produce
Enable
Processes
Decision Processes
Analytic Processes
Information Governance Processes
Platform
Decision Capabilities
Collaboration Decision Making, Intelligent Decision Automation, Applications
Analytic Capabilities
Descriptive, Diagnostic, Predictive, Prescriptive
Information Capabilities
Describe, Organize, Integrate, Share, Govern, Implement
Metadata and Services
Information

Gartner's Business Analytics Framework

Strategic BI starts from the business strategy

- Since the goal is to make sure that BI initiatives lead to decisions which improve the key business objectives, the fundamental requirement is the existence of a clearly defined business strategies and objectives.
- Typically, a business strategy and the key business goals are defined at a high, rather abstract level, and require further refinement to make it operational. Performance Management is a field which helps to further develop and execute the strategy and makes strategy more operational such that BI can be aligned with the business strategy.
- Performance Management consists of various techniques, such as:
  - Performance Prism (helps with interpretation of strategy)
  - Balance Score Cards (idem)
  - Profitability Modeling (Understanding what is driving your profit helps to finesse the strategy)
  - Strategy Map
- The goal of Performance Management is:
  - to link the enterprise strategic goals with operational activities
  - to relate Leading with Lagging metrics
  - to prevent silo tactical approaches leading to sub-optimization
  - to ensure that PM initiatives created at the intermediate level of the organization are linked to the business strategy.
  - to monitor strategy and support fact-based decision making

BI is about people

There are four types of participants:

- Consumer
  - Consumes the insights to make decisions and take actions that help the organization to achieve its goals
  - Originally, consumers were executives and managers, but there is trend where more and more operational workers are becoming consumers (e.g. call center handlers, inventory managers, retail outlet sales staff, ...)
  - **Consumers determine the success of BI initiatives**
- Producer
  - Analysts who carry out domain-specific and ad hoc analysis
  - Combines technical skills with deep understanding of business issues and performance measures.
- Enablers
  - Those who help to design, build and maintain the BI systems (Vendors, IT staff, BI Competence Centre)

  - Enablers need business knowledge and should be capable to work in multidisciplinary teams to ensure that the BI system creates value.
  - Enablers should not treat the business user as their customers, but involve them in the development by treating them as a team member.
- There is a fourth type appearing, called the prosumer
  - These are consumers which at the same time produce insights from data.
  - Prosumers are an important asset in any organization which constantly tries to link leading to lagging indicators.

BI is about processes

There are three types of processes:

- Business Processes
  - Gartner refers to these as decision processes, hereby referring to the decision making steps in a business process. We prefer to refer to business processes more generally.
  - The goal of business intelligence is to provide the right insights at the right time during the business process
  - In order to do so, business processes must embed BI-driven decision making.
- Analytical Processes
  - Analytical processes are the processes used to generate insights from data (by means of the BI system)
  - It is important to train producers in the analytical processes rather than the analytical tools
  - Most analytics processes which support strategic BI contain the following steps:
    - Define what you are trying to learn and the desired outcome of the analysis. Make sure it relates to the business strategy.
    - Define how to measure success of the business analytics exercise. This success should be linked to business results that matter.
    - Determine the required type of analysis, e.g. explorative, confirmatory, predictive or explanatory research. Identify appropriate techniques.
    - Identify the required data and models. Determine validity of the underlying assumptions and the available data.
    - Determine how the insights should be incorporated in the business process, e.g. process automation, suggestion generation, error prevention, providing relevant information and insights, …
- Information Governance Processes
  - Information Governance Processes are all the processes required to architect the BI system correctly and manage it once it is up and running.

BI is (also) about technology

The different technologies used comprise the BI platform. Again, three types of technological capabilities can be identified

- Decision capabilities
    - These are the capabilities which allow business users to better understand the business and make better decisions.
    - We can identify following categories of decision capabilities:
        - Information delivery: This is the core focus of most BI projects currently and relates to providing information to the decision maker.
        - Integration capabilities: This relates to technology which allows the decision maker to integrate different data sources to gain deeper insights into the context of the decision.
        - Analysis capabilities: This relates to technology which allows the decision maker to further analyze the provided data to make better decisions. An example is the use of scenario modeling and simulation to support flexible, iterative decision making
        - Intelligent decision automation: This provides support for structured hierarchical decisions.
- Analytical capabilities
    - These are the technological capabilities which allow producers to generate insights. Some examples are:
        - Text analytics
        - Content analytics
        - Speech analytics
        - Predictive analytics
        - Ad hoc queries
        - Visualizations
        - Statistics
        - Data mining
- Information capabilities
    - These are the capabilities which allow
        - for the creation of an information infrastructure that will unify all the existing information assets,
        - to get information out of their silos and
        - make the information available to the decision makers
- Note that the platform is only a part of the BI framework

BI should be supported by a Program Management

- When BI is applied from a strategic perspective, there will be multiple opportunities throughout the organization for BI initiatives, which all need to be aligned with the overall business strategy and integrated with each other.

- A BI Competence Center could take on the role of Program Management. At a minimum, they should be responsible for requirements gathering or needs analysis for BI within the organization and do so on an active, ongoing and iterative basis.
- An important part of the Program Management is also to prioritize BI initiatives. A portfolio approach could be useful here.

Metadata and Services

- Metadata is crucial to get data used across the organization
- Metadata management integrates the various metadata schemes and must also incorporate external sources
- Metadata must be shared through common repositories

The future of Business Intelligence

BI is dead, long live the new BI

Current and upcoming analytical capabilities of BI

- BI 1.0
  - These are the techniques most people think about when they hear the term Business Intelligence
  - Examples are:
    - Reporting
    - Dashboards
    - Ad Hoc Query
    - OLAP
    - Interactive Visualization
    - Data Mining
- BI 2.0
  - Since the early 2000, with the rise of Web 2.0, increasingly more data was collected about online customer behavior. This resulted in new capabilities which focusses on text and web-analytics for unstructured web content.
  - Examples are:
    - Text Mining
      - Information Extraction
      - Topic Identification
      - Opinion Mining
      - Question-Answering
    - Web Mining
    - Social Network Analysis

- BI 3.0
  - With the current rise of connected mobile devices and cheap sensors, resulting in the birth of Internet of Things (IoT), we are entering the Web 3.0 era (mobile and sensor-based).
  - Although BI 3.0 does not yet exist commercially, we can expect new analytical capabilities, such as:
    - Mobile analytics
    - Location and context-aware analytics
    - Big Data Analytics

Potential applications of BI

- E-commerce and market intelligence
  - Social Media monitoring and analysis
  - Towards long-tail, personalized marketing and increased customer satisfaction
- Smart health and well-being
  - From reactive, hospital-centered and focused on disease control
  - To Preventive, proactive, evidence-based, person-centered and focused on wellbeing
- Security and public safety
  - Terrorism informatics
  - Cyber security

Is BI Dead?

- Gartner, who introduced the concept BI, is proclaiming this year that BI is dead
  - Tools which require some form of IT intervention are no longer considered BI and no longer appear on Gartner's BI Magic Quadrant. Now they are called Enterprise-Reporting Based Platforms.
  - Gartner seems to have left the idea behind of trying to get a single view of data/truth.
  - They proclaim the time of BICC's planning BI initiatives is over.
  - Overall, according to Gartner there is a clear trend of IT-led BI towards BI-driven IT. New tools and technology make it possible to perform BI without IT.
- Reality is still different
  - According to a survey at a Gartner web seminar, the ideal result would be a mix of IT-led and business-led BI and the attendees did not believe IT-led would decrease.
  - IT clearly still has an important role to play in providing data to business people.

Towards Process-centric BI

- Traditional BI approaches are often user-driven and data-centered
  - It focusses on a specific user and the decision she has to make.

    - If often focusses on specific decisions in isolation of the preceding and succeeding decisions.
    - Often has a rather tactical nature.
    - It often starts with building a data warehouse and focusses on the required data structures to inform the consumer for making the appropriate decision.
- To really reap the benefits of BI, one should move towards a process-centric BI
    - BI inserts metrics, query results, analytical insights and other capabilities directly into the workflow of the business processes within the organization.
    - Starts from the entire process, instead of a specific decision/user.
- Process-driven BI has following implication for capabilities of a BI Platform:
    - Information delivery should be less focused on just providing reports, but the timing and the right kind of information provided becomes more important
    - Integration capabilities will have to incorporate concepts such as Service-oriented Architecture (SOA), Workflow Management Systems (WFMS), Complex Event Processing (CEP) and Business Rules to embed BI within a process.
- There are three dimensions to categorize Process-centered BI:
    - Process type:
        - Management processes
        - Business processes
        - Support processes
    - Process structuredness:
        - Well-structured
            - Predictable and definable inputs" min.)_
            - Linear logical workflow
            - Clearly definable tasks
            - Predictable and desired outcome
        - Unstructured
    - View of the data:
        - Narrow view: only rely on data from inside the process
        - Broad view: also rely on data from outside the process (could still be from within the organization!)
    - Adaptation of PCBI requires more effort when:
        - Going from support to business to management processes
        - Going from structured processes to unstructured processes
        - Going from a narrow view on data to a broad view on data

Resources

1. 12 simple rules: How Ted Codd transformed the humble database[1] *(1551 words, 6 min.)*

[1] http://www.theregister.co.uk/2013/08/19/ted_codd_90_relational_daddy/

2. Material Requirements Planning (MRP)[2] *(2800 words, 11 min.)*
3. Manufacturing Resource Planning (MRP II)[3] *(1300 words, 5 min.)*
4. Enterprise Resource Planning (ERP)[4] *(3200 words, 13 min.)*
5. Artificial intelligence[5] *(9700 words, 39 min.)*
6. Artificial intelligence is dead, long live deep learning[6] *(2300 words, 9 min.)*
7. The difference between data mining, statistics and artificial intelligence[7] *(600 words, 3 min.)*
8. History of Decision Support Systems[8] *(5900 words, 24 min.)*
9. Gartner's Business Analytics Framework[9] *(6500 words, 26 min.)*
10. Gartner, *Deliver Process-Driven Business Intelligence With a Balanced BI Platform*, April 2006[10] *(1600 words, 6 min.)*
11. Chen H., Chiang, R.H.L. and Storey, V.C., 2012. *Business Intelligence and Analytics: from Big Data to Big Impact.* MIS Quarterly, 36(4), pp. 1165-1188[11] *(13000 words, 52 min.)*
12. BI is dead[12] *(900 words, 4 min.)*
13. Tobias Bucher, Anke Gericke and Stefan Sigg, 2009. *Process-centric business intelligence.* Business Process Management Journal, 15(3), pp. 408-429[13] *(8500 words, 34 min.)*

[2] https://en.wikipedia.org/wiki/Material_requirements_planning

[3] https://en.wikipedia.org/wiki/Manufacturing_resource_planning

[4] https://en.wikipedia.org/wiki/Enterprise_resource_planning

[5] https://en.wikipedia.org/wiki/Artificial_intelligence

[6] http://www.kdnuggets.com/2016/08/artificial-intelligence-dead-long-live-deep-learning.html

[7] http://stats.stackexchange.com/a/21669

[8] http://dssresources.com/history/dsshistory.html

[9] http://www.gartner.com/imagesrv/summits/docs/na/business-intelligence/gartners_business_analytics__219420.pdf

[10] http://www.umsl.edu/~sauterv/DSS/pdf/BI/deliver_processdriven_busine_139377.pdf

[11] http://s3.amazonaws.com/academia.edu.documents/32970305/FROM_BIG_DATA_TO_BIG_IMPACT.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1474235157&Signature=trYMObns39i715eh92W0NeeQaUU%3D&response-content-disposition=inline%3B%20filename%3DSPECIAL_ISSUE_BUSINESS_INTELLIGENCE_RESE.pdf

[12] http://timoelliott.com/blog/2016/04/bi-is-dead.html

[13] https://pdfs.semanticscholar.org/0e5a/8b74064eb0f9f3f2f12b6a1f01b1ffd367df.pdf

Business Intelligence Architecture

Introduction to Software Architecture

The decisions made when designing the architecture are fundamental to what the system will become

Understanding Software Architecture

- A Software Architecture is a designed solution to solve a business problem or technical need.
- It identifies the different pieces of the solution and how they fit together
- It is a high-level solution, which abstracts from the details of implementation

Components of a Software Architecture

A Software Architecture covers among others the following aspects

- A description of the goals and purpose of the system
- The assumptions about its environment
- A collection of different views describing the parts of the system and how they relate

A Software Architecture contains multiple views

An architectural view represents a specific focus on the design and makes an appropriate abstraction of details irrelevant for the corresponding view. Some common views are:

- Logical View
  - Identifies the components of a system
  - Identifies the functionalities provided by each component
  - Identifies the relationships between the components
- Process View
  - Explains how the components of the system work together
  - Explains how the components of the system stay synchronized
- Physical View
  - Identifies the physical components of the system (network, servers, ...)
  - Maps the software to the computing platforms
- Development View
  - Describes how the software should be constructed at the development level
  - Shows how the different building blocks of the software are related and interdependent.

Software Architecture is about Requirements

- In order for an architecture to be a solution to a specific problem, it must meet several requirements.
- In software architectures, we typically distinguish between two types of requirements
  - Functional Requirements
    - Define what the system must do
    - Can be illustrated in a use case (if user does X, than system must do Y)
  - Non-functional Requirements
    - Cannot be demonstrated by 'clicking a button'
    - It doesn't describe what the system should do, but how it should behave/be
    - Examples are
      - Extensibility
      - Restructuring
      - Interoperability
      - Performance
      - Reliability
      - Availability
      - Maintainability
      - Security
      - Safety
      - Testability
      - Reusability
      - …
    - There often exists a trade-off between different non-functional requirements (e.g. performance vs security).
    - Selecting the appropriate trade-off is what architectural decisions are all about.

Software Architecture is about Abstractions

- Abstraction is an important principle in the creation of a software architecture. It relates to describing something in general terms that leaves out the details that are not relevant.
- Views are a first abstraction technique used in software architecture.
- Layers are a second abstraction technique which are used when building an architecture.
  - Layers describe an explicit set of functionalities
  - Layers can be created logically (software) or physically (hardware)
  - When layers are created physically, they are often called tiers
  - A common three-tier architecture is the 3-tier model, which separates the data, logic and presentation functionalities in their own layer.

Other Guiding Principles when Building a Software Architecture

- Modularization

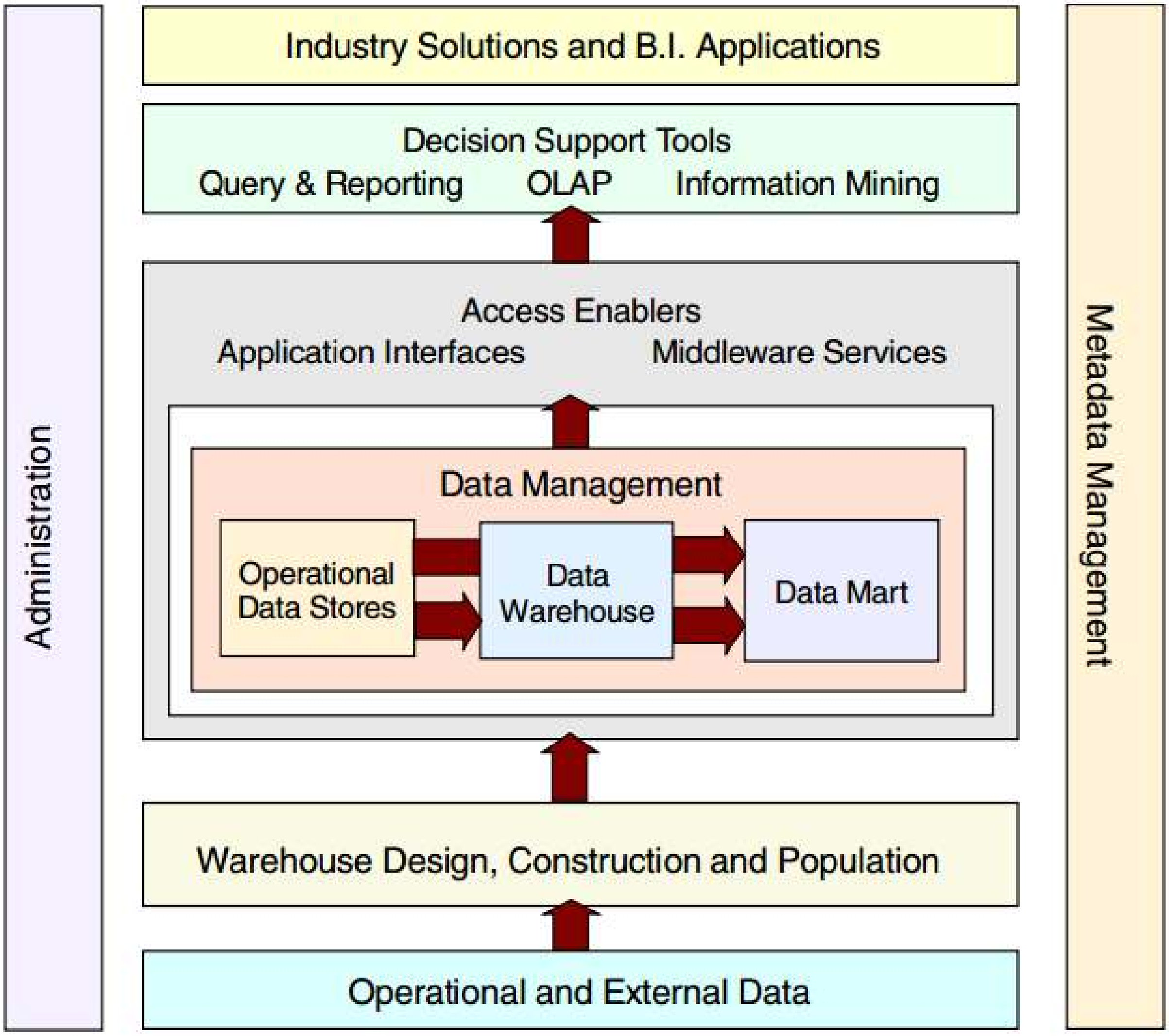
  - Break the system into parts with well-defined boundaries
- Separation of Concerns
  - Unrelated responsibilities should be separated in the system
- Coupling and Cohesion
  - High cohesion implies that functions within a component (module, layer, class, …) are strongly related to each other
  - Low coupling implies that components in a system are minimally related to other components
- These principles are important to realize certain non-functional requirements such as maintainability and restructuring

BI Architecture Basics

Need for BI

- Understanding your environment (competitors, customers, …) and your operations is becoming increasingly important in a highly competitive market.
- The data which are collected by companies are growing at an increasing rate.
- Some estimate that less than 10% of the captured data is actually used for decision-making.
- Goal of BI is to create actionable insights from data
- A Business Executive expects from a BI Architecture:
  - Application Freedom and unlimited access to data at all times
  - Information systems which are in sync with business processes
  - Low purchase cost
  - Transforming information into actions
- A CIO expects from a BI Architecture:
  - Accessible from all systems
  - Capable of dealing with heterogeneous data sources
  - Dynamic resource management
  - Non-stop availability

Top-level BI Architecture



IBM BI Architecture

Operational and External Data

- Also known as the Data Source Layer
- Can be structured, semi-structured and unstructured data
- Internal Data Sources
  - Online Transaction Processing (OLTP) Systems
  - Data used by systems whose main concern is to keep the business processes running.
  - Only keeps limited history of transactions (e.g. all open customer orders and closed customer orders up to three years)

- External data sources
    - Originates from outside the organization
    - Often relates to competitors, market, environment and technology
- Because these systems are not fully integrated and only keep a limited history, they are not well-suited to answer strategic business questions.

Warehouse Design, Construction and Population

- Also known as the ETL Layer
- ETL = Extract - Transform - Load
- Extract
    - Collects relevant data from different sources
    - Stores this in the data staging area (=temporary storage area)
        - Needed when ETL is too complex to do in a single pass.
        - Keeps interaction with the source systems to a minimum.
- Transformation
    - Clean the data using data quality rules
    - If the data is dirty according to the rule,
        - the data is rejected or
        - the data is corrected or
        - the data is allowed as-is
    - Data which does not meet data quality rules, are stored in a data quality database.
    - Transforms data to appropriate format and to allow integration of different data sources (e.g. conversions of monetary units).
- Loading
    - Loads the data from the staging area to the target repository
- Audit System logs all the operations done by the ETL system
    - Part of the ETL system.
    - Used for understanding what happened during the ETL process.
- Control System
    - Determines when each ETL component needs to run
    - Determines the sequence of the ETL processes
    - Allows ETL to restart from the point of failure
- Data Quality Database

Data Management

- Also known as the Data Warehouse Layer
- Consists of various components: operational data stores, data warehouse, data marts
- Operational Data Store (ODS)
    - Subject-oriented

        - Data store organized around a specific subject area (e.g. Customer Support)
    - Contains transaction-data
    - Integrated
        - Data from several data sources (e.g. order system, invoice system, CRM system, ...)
    - Updatable
        - Data can be added and updated in ODS
        - Data which comes from data sources, cannot be updated
    - Normalized Data Structure (3NF)
    - Consistent picture of current data
        - Does not store historical data
    - Real-time or Near-real-time updated
    - Hybrid
        - Used internally in DW Architecture, not accessible by user
        - User faced: available to end user
- Data Warehouse
    - Subject-oriented
        - Data store organized around a specific subject area
    - Contains summarized data
    - Integrated:
        - Data from several data sources
    - Time-variant
        - A Data Warehouse accumulates and stores historical changes
    - Non-volatile
        - Data is read-only, not updatable
    - User faced
        - Available to end user
    - Can either be a normalized or dimensional database
- Data Mart
    - A subset of the data warehouse
    - Supports analytical needs of a particular business function or department
    - Contains historical data up to a certain moment in time (limited lifespan of data)
    - Does not facilitate analysis across multiple business units
- Other common building blocks are
    - Multidimensional Database (MDB)
        - Database where data is stored in cells and the position of each cell is defined by a number of variables called dimensions
        - Special structure used for special applications such as multidimensional OLAP
    - Normalized Data Store (NDS)
        - Internally-focused
            - Not available to the end user
        - Normalized data structure

        - Purpose of integrating data from various source systems captured in a stage before it is loaded to a user-facing data store.

Data Management

- Database Engine
- Used to access and maintain information stored in DW Database

Access Enablers

- Provides user interfaces and middleware services which connect the end user to the DWH

Decision Support Tools

- GUI and Web-based BI tools and analytic applications
- Enables business end users to access and analyze DWH information
- Examples are:
    - Query and Reporting Tools. Often made available through BI portals
    - OLAP
        - Data manipulation engine designed to support multi-dimensional data structures
        - Roll-up or drill-up
        - Drill-down
        - Slice (selecting specific value on a single dimension) and Dice (selecting a range of values on two or more dimensions)
        - Pivot (rotate the axes)
    - Data Visualization (Dashboards and scorecards)

Industry Solutions and BI Applications

- Complete business intelligence solution packages tailored by industry or area of application
- Analytical Applications typically support following functions
    - Modeling
    - Forecasting
    - What-if analysis

Metadata Layer

- Administrative metadata
    - Provides administrators information about the structure of a DWH
    - Description of the source DBs
    - Definitions of the DWH Schemas

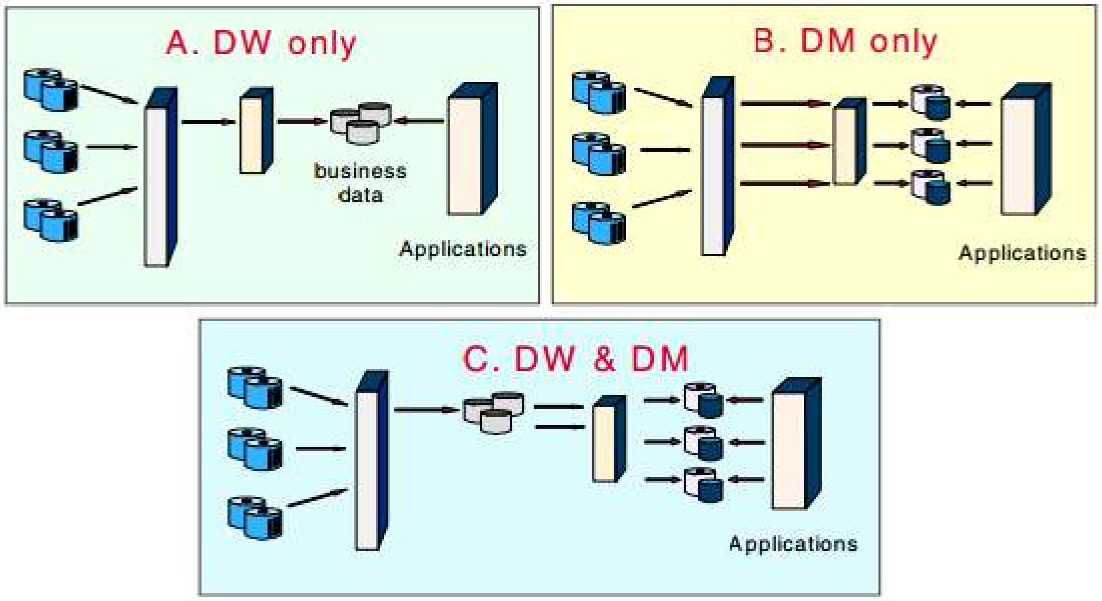
  - Dimensions and hierarchies
  - Predefined queries and reports
  - Data mart locations and content
  - Physical organization
  - ETL rules
  - Data refresh and purging policies
- Business metadata
  - Provides business users and BI producers information about the content and the meaning of information stored in the DWH
  - Business terms and definitions
  - Ownership of the data
  - Charging policies
- Operational metadata
  - Provides administrators information about the operations of a DWH
  - Usage statistics
  - Error reports
  - Audit trails

Administration

- Set of tools and services for managing data warehouse operations

Data Warehouse Architecture

Three Basic DWH Architectures



Three basic DWH architectures

DW Only

- Based on a single enterprise-wide DWH
- Single centralized location for BI data
- Strengths
  - Stores data only once
  - Data duplication and storage requirements are minimized
  - Data is consolidated enterprise-wide
- Weaknesses
  - Data is not structured to support the individual informational needs of specific end users or groups
  - Requires that the entire organization accepts the same definitions and transformations of data

DM Only

- Each department has its own DM
- Strengths
  - Data Marts can be deployed very rapidly

  - Specifically responsive to identified business problems
  - Optimized for the needs of particular groups of users, which makes population of the DM easier and may also yield better performance
- Weaknesses
  - Data is stored multiple times in different data marts (data duplication, increased data storage requirements)
  - Data is not consolidated enterprise-wide, no corporate view of business information

DW and DM

- DWH represents the central location of consolidated enterprise data (single version of the truth)
- Data Marts represent departmental slices of the data
- Strengths
  - Data mart creation and population are simplified because data can be extracted from enterprise DWH
  - Data marts are in sync and compatible with the enterprise view
- Weaknesses
  - Data duplication, increased data storage requirements
  - Requires that the entire organization accepts the same definitions of data

Bill Inmon

- Father of Data Warehousing
- Coined the term DW in 1990 with his book 'Building the Data Warehouse'
- Considers the DWH as an integral part of the Corporate Information Factory (CIF)
  - Environment consisting of all information systems and their databases throughout a given organization
  - DW and operational databases are part of a larger whole
- Primary audience: IT Professionals
- Overall Approach: top-down
  - Define a enterprise-wide logical architecture
  - Extract and transform data accordingly from operational databases into a single database (DWH)
  - Extract from Enterprise DWH data to create smaller departmental databases
  - Individual users in turn can extract from the departmental databases to create their ad hoc data sets to support their individual decision processes
- Based on Normalization

Ralph Kimball

- Publishes his own book in 1996, called 'The Data Warehouse Toolkit'
- Suggests a drastically different approach
- Considers the DWH as a transformer and retainer of operational data
- Primary audience: End users
- Based on dimensional modeling
- Top-down approach:
  - Enterprise Data Warehouse Bus Architecture
  - Develop the Enterprise Data Warehouse Bus Matrix
    - Rows represent business processes
    - Columns represent common dimensions used across the enterprise
    - Identify which processes (dimensional models) use which conformed dimensions.
  - Develop conformed dimensions once with all relevant stakeholders
  - Define conformed facts
- Bottom-up approach:
  - Select a business process
  - Define the grain
  - Choose the dimensions
  - Identify the facts

Resources

1. Pattern-Oriented Software Architecture For Dummies, chapters 1 and 2[14] *(8200 words, 50 min.)*
2. Business Intelligence Architecture on S/390[15] *(2400 words, 16 min.)*
3. A Five-Layered Business Intelligence Architecture[16] *(4300 words, 30 min.)*
4. An Overview of Data Warehousing and OLAP Technology[17] *(7400 words, 50 min.)*
5. Data Warehousing. Battle of the Giants: Comparing the Basics of the Kimball and Inmon Models[18] *(8300 words, 55 min.)*
6. Which Data Warehouse Architecture Is Most Successful[19] *(2400 words, 16 min.)*
7. The Data Warehouse Toolkit by Margy Ross and Ralph Kimball. (chapter 4, sections 'Enterprise Data Warehouse Bus Architecture', 'Conformed Dimensions', 'Conformed Facts')[20] *(8000 words, 50 min.)*

[14] https://ebookcentral-proquest-com.bib-proxy.uhasselt.be/lib/ubhasselt/reader.action?ppg=62&docID=1095125&tm=1474839114786
[15] http://www.redbooks.ibm.com/redbooks/pdfs/sg245641.pdf
[16] http://www.ibimapublishing.com/journals/CIBIMA/2011/695619/695619.pdf
[17] https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/sigrecord.pdf
[18] http://olap.it/Articoli/Battle%20of%20the%20giants%20-%20comparing%20Kimball%20and%20Inmon.pdf
[19] https://cours.etsmtl.ca/mti820/public_docs/lectures/WhichDWArchitectureIsMostSuccessful.pdf
[20] https://ebookcentral.proquest.com/lib/ubhasselt/detail.action?docID=1313513

Data Warehouse Architecture

Dimensional Modeling Fundamentals

Four steps are all you need to master, to become a dimensional modeler.

Step 1: Select the Business Process of Interest

- To start, focus on the business process which is the most critical to the business users as well as the most feasible.
- Hints to identify the activity of interest:
  - These processes are typically supported by an operational system
  - These processes generate or capture key performance metrics
  - A business function/department IS NOT a business activity/process

Step 2: Declare the grain

- Specify what a single row in the fact table represents (grain)
  - Define the grain in business terms, not as a set of foreign keys (whenever possible!)
  - Define the grain at the lowest atomic level possible. This will result in data which is highly dimensional and provides maximum analytic flexibility
- Determine the type(s) of fact table you want/need
  - Transaction Fact Table
  - Periodic Snapshot Fact Table
  - Accumulating Snapshot Fact Table
  - Factless Fact Table
  - Often a combination of these types are needed to get a full picture of the process.
- The most frequent error is not declaring the grain of the fact table from the start

Step 3: Identify the dimensions

- Dimensions are how do business people describe the data resulting from the business process measurement events
- Look for the 'Who, what, where, when, why and how' associated with the facts
- Determine the strategy for slow changes in the dimensions
  - Type 0: retain original
  - Type 1: overwrite

  - Type 2: add row
  - Type 3: add column
  - Type 4: mini-dimension
  - Type 5: Mini-dimension and type 1 outrigger
  - Type 6: Add type 1 attributes to type 2 Dimension
  - Type 7: Dual type 1 and type 2 dimensions

Step 4: Identify the Facts

- What are the facts about the process you are interested in?
  - Facts are often metric measurements recorded in the process.
  - Facts are typically additive, i.e. it makes sense to add theese values for a set of rows, grouped according to a dimension (e.g. total 'amount in euro' for all sales over a period of time makes sense)
  - To identify the relevant facts, you should consider both the source data as the business reality
- All facts must be true to the grain, if not they belong in a separate fact table
- In general derived facts should be stored, not calculated
  - This way they are consistently computed by ETL and there is no risk of a user incorrectly calculating a derived fact
  - An alternative is to use a View instead, but this requires that there is no way the user can bypass the view
- Non-additive facts have little value and one should be careful when summed or averaged
  - Non-additive facts are measurements which you cannot add across a dimension, as the result doesn't make sense. (e.g. total 'unit price' of all sales over a period of time makes no sense)
- Semi-additive facts can be useful, but must be treated with caution.
  - A semi-additive fact is additive for some dimensions, while non-additive for other dimensions. (e.g. total 'amount due' by customers over a period of time makes no sense, but total 'amount due' for all customers on a specific date does make sense)

Fact Tables

Intelligent decision making is fact-based.

Transaction Fact Table

- The fact grain represents the actual transactions (e.g. a row for each product sold to a specific customer on a specific date)
- A fact table can contain different types of transaction (e.g. products sold over the counter, products sold online, products sold with invoice, …). There are two ways to deal with this:

  - Create one fact table and use a dimension to identify the transaction type
  - Create separate fact tables for each transaction type
- To make the design decision whether you build a blended transaction fact table with a transaction type dimension or to build a separate fact table for each transaction type, you should take following elements into consideration
  - Which design matches the analytic requirements of the business users the best?
  - Are there really multiple unique business processes? If so, go for separate fact tables.
  - Are multiple source systems capturing metrics with unique granularities? If so, this suggests separate fact tables.
  - Do the dimensionalities associated with the facts differ? This suggests again separate fact tables.

Periodic Snapshot Fact Table

- The fact table grain represents a static status level, which is the result of consecutive transactions. E.g. the inventory level for a specific product on a given day.
- If the purpose is to keep a daily snapshot, every day new facts are added to the fact table representing the new static level.
- Note that facts do not represent transactions, but the status of the system (which changes under the influence of transactions)
- Periodic snapshot fact tables can be dense (a lot of rows for all possible combinations of the dimension rows), which will lead to an explosion of the number of fact rows.
  - A tactic is to keep the daily snapshot fact table up to date for the past 60 days and for more history one keeps a monthly periodic snapshot fact table (= different grain).
- Often, the facts of a periodic snapshot table are semi-additive, i.e. they are not additive over the date dimension (e.g. quantity on hand), while fully additive over the other dimensions
- These facts which represent static levels can be averaged though over the date dimension
  - However, you can't use the SQL average function directly as this will divide over the number of rows and not the number of days (because you have multiple rows for each product for a give day)
- One can also add other facts to the same fact table, as long as they represent a static level, e.g. quantity sold (on that specific day for that specific product in that specific store)
- These fact tables are used to see cumulative performance of the business at regular, predictable time intervals
- The main place to easily retrieve a regular, predictable view of longitudinal performance trends

Accumulating Snapshot Fact Table

- The fact grain represents the lifecycle of the process by recording facts about different fixed milestones.

- This approach requires processes which have a definite beginning and end and identifiable milestones in between.
- Example for inventory level, described by product, warehouse and vendor
  - Milestones are: date received, date inspected, date bin placement, date initial shipment date last shipment
  - For each milestone a fact is recorded: quantity received, quantity inspected, quantity placed in bin, quantity shipped to customer
  - Not every fact should be accompanied by a specific date (when not deemed necessary), e.g. quantity returned to vendor is recorded without recording the explicit date
  - Each milestone date refers to a logical copy of the date dimension
- Because many of the milestone dates are unknown when the fact row is created, a default surrogate date key is used for the undefined dates
- Every time a transaction occurs which influence the current state of the process, an update has to be done of the fact row.
- Sometimes an additional date column is added which indicates when the snapshot row was last updated
- Also a status dimension can be added to represent the current status of the fact row.
- One can also add metrics (facts) which represent durations or lags between milestones. These can be simply the raw difference or more complicated by e.g. counting the number of workdays.
- These type of fact tables are most appropriate when business users want to perform workflow analysis

Keys and Null Values

- It is advisable to add a surrogate primary key to the fact table
  - Even if a combination of foreign keys would suffice as primary key.
  - Always add the suffix ‘PK’ to the primary key name.
- Always add the suffix ‘FK’ to the foreign keys (which refer to the dimensions)
- Sometimes, a fact (row in the fact table) might not have a corresponding row in one of the dimensions (e.g. online purchase has no corresponding row in the store dimension)
  - Never add NULL as value for the foreign key for those cases!
  - Instead, create a dummy dimension row in the dimension table (e.g. a row representing ‘not applicable’ with a key value of 0 or -1)
- Numeric facts which have a null value should remain null
  - Replacing these values with 0 for example will mess up the results of functions such as sum(), count() and average()

Factless Fact Tables

- Sometimes all we want to know are the existing combinations of dimension rows without having any fact measurements.

- In these cases, the fact table is factless
- An example is a fact table to know which store (dimension), ran which promotion (dimension), on which date (dimension) for which product (dimension)

Dimension Tables

To understand facts, one needs dimensions

Date Dimensions

- This dimension appears in nearly every dimensional model
- Typically the grain of the date dimension is a single day
- The date dimension can be build in advance
- The date dimension should contain many attributes, such as fields for both long and abbreviated date labels (01/01/2012 and '01 January 2012')
- The use of a date dimension has following benefits instead of storing dates directly as a fact are:
  - Not all users are fluent in SQL date semantics
  - SQL Date functions do not support filtering by attributes such as weekdays vs weekends
- Also useful are flag indicators in the date dimension to identify the current date
- Note that the date dimension is often defined at the level of a single day. If it is necessary to measure datetime at a finer grain, there are several options:
  - One could store time-of-day as a separate date dimension to avoid row count explosion in the date dimension
  - If there is no need to roll up or filter on time-of-day groupings, time-of-day should be handled as a simple date/time fact in the fact table
  - Since time is continuous, one still needs to define the grain. Often one does not need to go to the level of seconds or lower and a grain at the level of 15 minutes could be sufficient.

Flattened Hierarchies in Dimension Tables

- Dimension tables often contain many-to-one hierarchies: many products belong to one brand, many brands belong to one category, many categories belong to one department …
- Traditional data modeling would force you to normalize these relations in separate tables.
- In dimensional modeling, you should resist this normalization urge and allow repeated values.

Special Dimensional Attributes

- Flag and indicator attributes
  - These attributes represent a limited set of categorical values
  - Instead of using True and False as values for binary flag attributes, it is better to use meaningful labels such as ‘weekday’ and ‘weekend’
- Natural keys
  - These are identifiers which are often find in the operational databases and which contain embedded meaning
  - You should identify natural keys in the dimension table by using the suffix ‘NK’ in the attribute’s name.
- Numeric attributes
  - Typically, numeric values represent facts and go in the fact table. This is certainly the case when these numeric values are used for calculation purposes.
  - However, sometimes numeric values can be used predominantly for filtering and grouping, in which case they should be treated as a product dimension attribute
  - If the numeric values serve both purposes, you should duplicate them in the fact and dimension tables.
- Date Attributes
  - Sometimes dimensional attributes represent dates, e.g. first_open_date of a warehouse in the warehouse Dimension
  - If you want to group and filter on nonstandard calendar attributes (e.g. fiscal period, weekday/weekend), then you should not store the actual date, but a foreign key to a **copy** of the date dimension table
  - Note that these copies are not physical copies, but could be created as logical copies by means of a view.
  - Such a dimension which is linked to another dimension is called an ‘outrigger’.

Separating dimension attributes in separate dimensions

- Sometimes, you want to describe the facts by a combination of attributes which are related: e.g. describe sales by four promotion actions such as *price reduction*, *ads*, *displays*, and *coupons*
- One natural approach is to combine these four attributes in one dimension.
- A second approach is to create a separate dimension for each attribute
- Advantages and Disadvantages
  - If the four attributes are highly correlated, the single dimension is not much larger than any one of the separate dimensions
  - The single dimension reveals very efficiently how the various price reductions, ads, displays, and coupons are used together
  - The separate dimensions may be more understandable to the business if users think of these mechanisms separately
  - Separating dimension attributes in their own dimension can create a so-called centipede fact table, containing too much dimensions

Degenerate Dimensions

- Sometimes, it appears that a dimension only consists of its key. In other words, the dimension is empty
- In those cases, the dimension table is left out of the model but the key remains in the fact table.
- Such key is referred to as the degenerate dimension.
- A degenerate dimension key is still useful as it still acts as a grouping variable for various facts (e.g. assume the fact grain are the sales facts for each product in a POS transaction, then the POS transaction ID groups all sales facts which occurred in the same POS transaction)

Keys and Null Values

- The primary key of a dimension table should be a surrogate key, not the natural key from the operational systems.
- Use PK and FK as suffixes for primary keys and foreign keys
- Typically, the surrogate key has no meaning. One possible exception are the surrogate keys of the date dimension. Commonly, the primary key of the date dimension is an integer formatted as 'yyyymmdd'.
- Dimension attributes should not have null values, but values such as 'Unknown' or 'Not Applicable' as this is easier for the users to understand.

Slowly changing dimensions

No matter how slow dimensions might change, you need a strategy to deal with it

Slowly changing dimension basics

- Dimension tables are relatively static, but aren't fixed forever
- Dimension attributes values change over time, albeit rather slowly
- Designers must determine in advance the strategy to deal with these changes.

Type 0: Retain Original

- If a dimension attributes follows the type 0 strategy, it is never updated and always keeps its original value.
- This type is appropriate for any attribute labeled "original".
- The durable supernatural key is a type 0 attribute.

Type 1: Overwrite

- Overwrite the old attribute value in the dimension row, replacing it with the current value.
- The attribute always reflect the recent value.
- A type 1 attribute does not retain its history.
- A type 1 response is appropriate if the attribute change is an insignificant correction or if there is no value in keeping the old value.

Type 2: Add New Row

- The most common and safest technique to deal with slowly changing dimension attributes.
- Requires the following attributes to support type 2 changes: 'row effective date', 'row expiration date', 'current row indicator'.
- A new row is added with the current values, while the old row is retained.
- The 'row expiration date' of the old row and 'row effective date' of the new row are set to the current date.
- The 'row expiration date' of the new row is set to a very far date in the future ('e.g. 31/12/9999').
- The 'current row indicator' attribute of the old and new row is set respectively to 'expired' and 'current'.
- A natural key attribute (or durable supernatural key) is required to group the rows representing the history of the changes.
  - If the natural key can change over time, we need to create an artificial natural key which never changes, to keep the history of changes on a row together. This artificial natural key is the supernatural key.
- With type 2 changes, the fact table remains untouched. Old facts should remain connect to the old dimension rows, while new facts will be connected to the new dimension rows.

Type 3: Add New Attribute

- Type 3 is useful when you want to be able to analyze the fact data in a scenario as if the change never happened and the value had always been the current (new) value.
- A new column is added called with the prefix "Prior" in the attribute name: e.g. if you need to change the attribute "department", you add a new attribute "prior department".
- The prior field gets the old value, while the actual field is overwritten.
- If there is a predictable pattern of changes to an attribute of type 3, one sometimes adds multiple new attributes to keep a longer history than the previous year (e.g. 'department 2015', 'department 2014', 'department ...')

Type 4: Add Mini-Dimension

- When rate of changes speeds up, a type 2 approach is less appealing as the dimension table could blow up in size
- Solution is to break off frequently changing attributes into a separate dimension, referred to as a mini-dimension. E.g. a mini-dimension for more volatile demographic attributes such as age, purchase frequency and income level.
- The mini-dimension contains one row for each combination of attribute values, creating (demographic) profiles.
- Note that this requires discretization of the attributes. If users insist on access to a specific raw data value, it should be included in the fact table, in addition to being value banded in the demographic mini-dimension
- The fact table refers dirctly to the mini-dimension. This way, the fact table captures the demographic profile changes.
- The attributes in the mini-dimension are no longer directly connected to the dimension the attributes belonged to originally!

Type 5: Mini-Dimension and Type 1 Outrigger

- Similar to type 4, but now the mini-dimension is also connected to the original dimension (in addition to its connection to the fact table), by means of a type 1 foreign key (overwrite).
- Useful when you want to analyze historical facts based on current value of volatile attributes.
  - To analyze with current values, you look at the rows in the mini-dimension referenced to from the original dimension.
  - To analyze with the original values, you look at the rows in the mini-dimension referenced to from the fact table.

Type 6: Add Type 1 Attributes to Type 2 Dimension

- Similar to a type 2 attribute, with the three additional attributes 'row effective date', 'row expiration date' and 'current row indicator'
- Combined with a type 3 attribute with 'current' and 'prior' prefixes.
- The 'current'-attribute is a type 1 attribute (overwritten)
- If you want to group facts based on the attribute value that was in effect when the facts occurred, you use the 'historic' attribute
- If you want to group facts based on the current attribute value, you use the 'current' attribute value

Type 7: Dual Type 1 and Type 2 Dimensions

- Similar to a type 2 approach, where new rows are added for dealing with changing dimension values.
- In addition a View of the dimension is created containing all rows for which 'current row indicator' is set to 'current'
- The fact table connects to this View through the natural (durable!) key instead of the surrogate key
- The type 2 dimension keeps track of the history, while the View allows for analyzing the facts against the most recent values.

ETL Subsystems

Major components of every ETL architecture

- Extracting
  - Gathering raw data from various source systems
  - Typically writing it to disk
  - No significant restructuring takes place
- Cleaning and conforming
  - Series of processing steps
  - Improving the quality of the data
  - Merging data from different sources
  - While maintaining an "objective" reflection of reality
- Delivering
  - Physically structuring the data
  - Physically loading the data in the target dimensional models
- Managing
  - Managing the ETL systems and processes

Extracting Components

1. Data Profiling System
   - Generates profiling reports about:
     - Content of the data
     - Consistency of the data
     - Structure of the data
2. Change Data Capture System
   - Isolate the changes in the source data
   - This will make the ETL system more efficient
3. Extract System
   - Extract to filesystem
   - Extract as a stream

Cleaning and Conforming Components

1. Data Cleansing System
    - Find proper balance between cleaned data and data reflecting reality
    - Quality Screens
        - Diagnostic filters which act as a test for the data
        - Three types: column screens, structure screens and business rule screens
        - If data fails the test, there are three options
            * Halting the ETL process
            * Sending the offending records to a suspense file for later processing
            * Tagging the data and passing it through the next step in the ETL process (preferred)
2. Error Event Schema
    - Centralized dimensional schema
    - Records every error event thrown by a quality screen in the ETL process
3. Audit Dimension Assembler
    - Additional dimension to which each fact refers
    - Contains audit information about the record when it was created in the fact table
4. Deduplication System
    - (Partially) Matching data from different source systems
5. Conforming System

Delivering Components

1. Slowly Changing Dimension Manager
    - Deals with changing dimensional data
    - Determines appropriate SCD Type
    - Makes necessary transformations
2. Surrogate Key Generator
    - Should generate surrogate keys for every dimension independently
3. Hierarchy Manager
4. Special Dimensions Manager
5. Fact Table Builders
    - First build the dimensions, next the facts!
6. Surrogate Key Pipeline
    - Replace the operational natural keys in the fact table by the appropriate surrogate keys
7. Multivalued Dimension Bridge Table Builder
8. Late Arriving Data Handler
9. Dimension Manager System
10. Fact Provider System
11. Aggregate Builder
12. OLAP Cube Builder
13. Data Propagation Manager

Management Components

1. Job Scheduler
2. Backup System
3. Recovery and Restart System
4. Version Control System
5. Version Migration System
6. Workflow Monitor
7. Sorting System
8. Lineage and Dependency Analyzer
9. Problem Escalation System
10. Parallelizing/Pipelining System
11. Security System
12. Compliance Manager
13. Metadata Repository Manager

Resources

1. The Data Warehouse Toolkit by Margy Ross and Ralph Kimball. (chapters 3-6 + 19)[21] *(70000 words, 480 min.)*

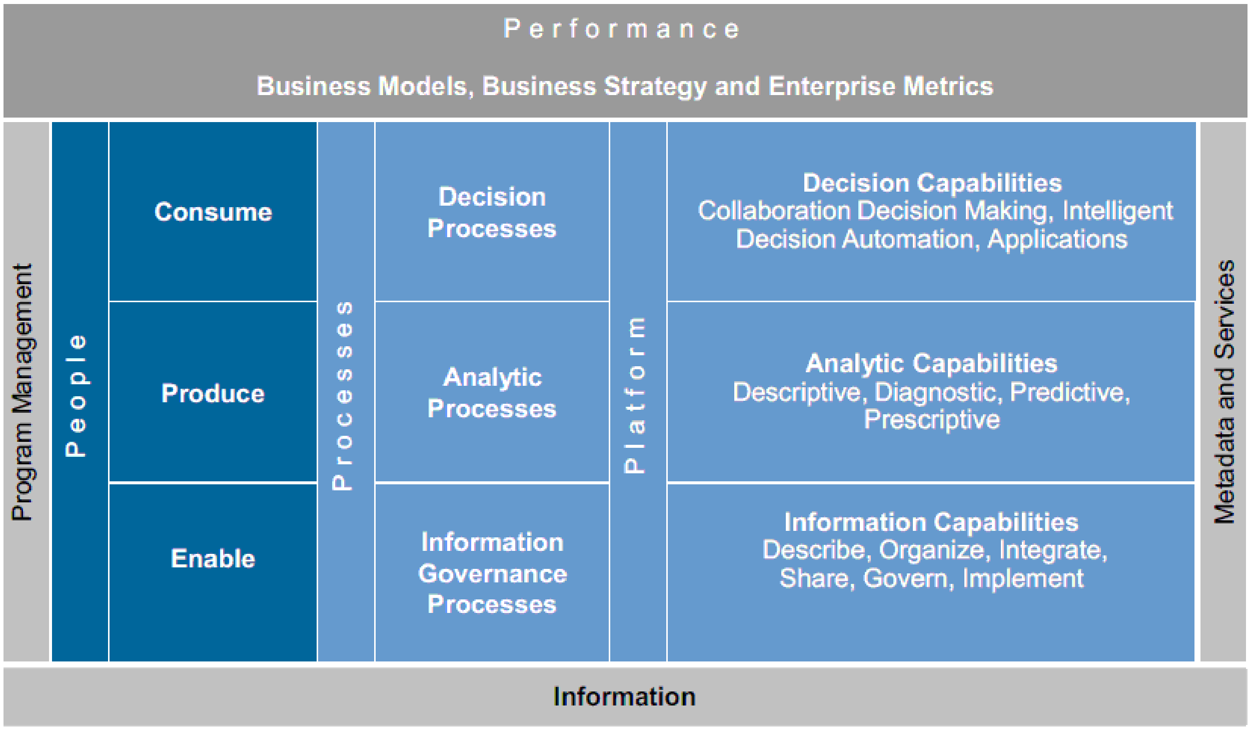
[21] https://ebookcentral.proquest.com/lib/ubhasselt/detail.action?docID=1313513

Business Intelligence, Business Strategy and Performance Management

Recap

A BI Framework

- If a company wants to use Business Intelligence strategically, it requires a framework to fully understand the different elements of strategic BI.
- For this course, we will use Gartner's Business Analytics Framework, which identifies three main components which should be aligned and integrated:
  - People
  - Processes
  - Platforms
- The link between the BI initiative and the organization's strategy is realized by the *Performance* component.


Performance
Business Models, Business Strategy and Enterprise Metrics
Program Management
People
Consume
Produce
Enable
Processes
Decision Processes
Analytic Processes
Information Governance Processes
Platform
Decision Capabilities
Collaboration Decision Making, Intelligent Decision Automation, Applications
Analytic Capabilities
Descriptive, Diagnostic, Predictive, Prescriptive
Information Capabilities
Describe, Organize, Integrate, Share, Govern, Implement
Metadata and Services
Information

Gartner's Business Analytics Framework

Strategic BI starts from the business strategy

- Since the goal is to make sure that BI initiatives lead to decisions which improve the key business objectives, the fundamental requirement is the existence of a clearly defined business strategies and objectives.
- Typically, a business strategy and the key business goals are defined at a high, rather abstract level, and require further refinement to make it operational. Performance Management is a field which helps to further develop and execute the strategy and makes strategy more operational such that BI can be aligned with the business strategy.
- Performance Management consists of various techniques, such as:
  - Performance Prism (helps with interpretation of strategy)
  - Balance Score Cards (idem)
  - Profitability Modeling (Understanding what is driving your profit helps to finesse the strategy)
  - Strategy Map
- The goal of Performance Management is:
  - to link the enterprise strategic goals with operational activities
  - to relate Leading with Lagging metrics
  - to prevent silo tactical approaches leading to sub-optimization
  - to ensure that PM initiatives created at the intermediate level of the organization are linked to the business strategy.

- to monitor strategy and support fact-based decision making

Critical Success Factors

Historical Background

- The term Critical Success Factor (CSF) was introduced by John F. Rockart in his 1979 article in the Harvard Business Review, called 'Chief executives define their own data needs'
- In this article, he describes the challenge executive managers back then faced of lacking an clear picture of the companies performance despite the many reports they received.
- He identifies four approaches how executive managers receive 'relevant' information
  - By-product approach
    - All reports the executive manager receives are by-products of particular systems designed to perform routine paperwork processing.
    - Drawback is that those by-products are often not very useful to an executive manager as they are typically created in isolated functional silo's.
  - Null approach
    - Because it is impossible to determine which information an executive needs, he fully relies on ad-hoc, rapidly assembled and often subjective information delivered by word of mouth of trusted advisers
    - This approach reflects the research of Mintzberg which showed that most managers rely on soft and speculative impressions, while they often ignored the hard data.
    - This approach completely ignores the management-by-fact paradigm
  - Key indicator system approach
    - A set of key indicators of the organization's health are selected
    - The results of these indicators are only shown when they deviate from the expected value
    - The results are visually displayed to the executive
    - This approach is in fact the predecessor of the modern day business dashboards.
    - Back in the days of Rockart, the indicators were mainly financial.
    - Drawback of this approach is that it typically lacks a real focus and mainly deals with hard (financial) information.
  - Total Study approach
    - Subsystems necessary to provide relevant information currently unavailable are identified and assigned priorities for development
    - Top-down analysis of the information needs
    - Many managers are interviewed to determine environment, objectives, key decisions and information needs
    - The objective is to develop an overall understanding of the business and its information needs
    - Drawback of this approach is that it is very expensive and not very agile. * ### Critical Success Factors

- A CSF is a factor which determines success, i.e. the key jobs which must be done exceedingly well for a company to be successful
    - Areas in which results, if they are satisfactory, will ensure successful competitive performance
    - The key areas where "things must go right" for the business to flourish
    - These areas should receive constant and careful attention from management
    - Others define CSF as "things which need to be done" in order for the organization to be successful, rather than "areas".
- Note that Rockart makes a difference between CSF and organizational goals
    - Goals represent the end points that an organization hopes to reach (e.g. 30% market share)
    - CSF are the areas in which good performance is necessary to ensure attainment of those goals. (Strong product mix, efficient sales promotion)
- One can identify two types of CSF
    - Monitoring CSF
        * These CSF deal with monitoring the current results
        * It is critical for the company's success to maintain its current level of results
    - Building CSF
        * These CSF deal with building for the future
        * It is critical for the company's success to realize some important changes

The CSF Approach

- Rockart introduces a new approach, based on what he calls critical success factors
    1. Record the executive's goals
    2. Determine the underlying CSF
    3. Discuss, clarify and determine the final set of CSF
    4. Define measures relevant for each CSF
    5. Create a control system (IT and processes) which reports on the CSF to the executive managers.
- The CSF approach focuses on individual managers and their current information needs, both hard and soft.
- CSF should be defined hierarchical, top-down and in relation to the company's mission and strategy.
    1. Define CSFs for the overall organization, aimed at fulfilling the corporate mission.
    2. Define CSFs for each business unit which reflect the specific business environments, while supporting the overall corporate objective.
    3. Define CSFs for each function within a business unit.
- Only five to ten CSFs should be defined at each level.

Balanced Score Card and Strategy Maps

Must read

1. Kaplan, R.S., Norton, D.P., 1993. *Putting the balanced scorecard to work.* Harvard Business Review, September-October.[22] *(8000 words, 55 min.)*
2. Kaplan, R.S., Norton, D.P., 2000. *Having trouble with strategy? Then Map It.* Harvard Business Review, September-October.[23] *(4400 words, 30 min.)*

Performance Prism

Must read

1. Neely, A., Adams, C., Crowe, P., 2001. *The performance prism in practice.* Measuring Business Excellence, 5(2), pp. 6-12.[24] *(4800 words, 30 mins.)*

Key Performance Indicator Modeling

Must read

1. Strecker, S., Frank, U., Heise, D., Kattenstroth, H., 2012. *MetricM: A modeling method in support of the reflective design and use of performance measurement systems.* Information Systems and e-Business Management, 10(2), pp. 241-276.[25] *(16600 words, 110 min.)*

Resources

1. Rockart, J.F., 1979. *Chief executives define their own data needs.* Harvard Business Review, March-April, pp. 81-93.[26] *(9000 words, 60 min.)*
2. Freund, Y.P., 1988. *Critical success factors.* Planning Review, 16(4), pp. 20-23.[27] *(2000 words, 15 min.)*

[22] http://s3.amazonaws.com/academia.edu.documents/43662394/Kaplan_Norton_Balanced_Scorecard_-_3_articles.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1478562082&Signature=NvzbvmKpeX7anhvm1V3aEmj0s0w%3D&response-content-disposition=inline%3B%20filename%3D2_Putting_the_Balanced_Scorecard_to_Work.pdf#page=3

[23] http://s3.amazonaws.com/academia.edu.documents/43662394/Kaplan_Norton_Balanced_Scorecard_-_3_articles.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1478562082&Signature=NvzbvmKpeX7anhvm1V3aEmj0s0w%3D&response-content-disposition=inline%3B%20filename%3D2_Putting_the_Balanced_Scorecard_to_Work.pdf#page=3

[24] https://www.researchgate.net/profile/Andy_Neely/publication/228602984_The_performance_prism_in_practice/links/0deec53236ff198557000000.pdf

[25] https://www.fernuni-hagen.de/evis/download/forschung/strecker-iseb.pdf

[26] https://scholar.google.be/scholar?hl=nl&q=Chief+executives+define+their+own+data+needs

[27] https://scholar.google.be/scholar?q=Critical+success+factors

3. Kaplan, R.S., Norton, D.P., 1993. *Putting the balanced scorecard to work.* Harvard Business Review, September-October.[28] *(8000 words, 55 min.)*
4. Kaplan, R.S., Norton, D.P., 2000. *Having trouble with strategy? Then Map It.* Harvard Business Review, September-October.[29] *(4400 words, 30 min.)*
5. Neely, A., Adams, C., Crowe, P., 2001. *The performance prism in practice.* Measuring Business Excellence, 5(2), pp. 6-12.[30] *(4800 words, 30 mins.)*
6. Strecker, S., Frank, U., Heise, D., Kattenstroth, H., 2012. *MetricM: A modeling method in support of the reflective design and use of performance measurement systems.* Information Systems and e-Business Management, 10(2), pp. 241-276.[31] *(16600 words, 110 min.)*

[28] http://s3.amazonaws.com/academia.edu.documents/43662394/Kaplan_Norton_Balanced_Scorecard_-_3_articles.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1478562082&Signature=NvzbvmKpeX7anhvm1V3aEmj0s0w%3D&response-content-disposition=inline%3B%20filename%3D2_Putting_the_Balanced_Scorecard_to_Work.pdf#page=3

[29] http://s3.amazonaws.com/academia.edu.documents/43662394/Kaplan_Norton_Balanced_Scorecard_-_3_articles.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1478562082&Signature=NvzbvmKpeX7anhvm1V3aEmj0s0w%3D&response-content-disposition=inline%3B%20filename%3D2_Putting_the_Balanced_Scorecard_to_Work.pdf#page=3

[30] https://www.researchgate.net/profile/Andy_Neely/publication/228602984_The_performance_prism_in_practice/links/0deec53236ff198557000000.pdf

[31] https://www.fernuni-hagen.de/evis/download/forschung/strecker-iseb.pdf