Relacja *TempNoon(Patient, Value, Time, Date)* dla każdego pacjenta i dnia zawiera pomiary temperatury około południa, może być rozpatrywana jako zmaterializowany widok instancji w Tabeli 6.

Zgodnie z (3) możemy użyć następującej definicji widoku, która uwzględnia pomiary wykonane pomiędzy 11:30, a 12:30.

(5)

Materializując powyższy widok otrzymamy instancję pokazaną w Tabeli 1.

W celu wyrażenia oczekiwań w stosunku do jakości, wprowadzone zostaną predykaty jakości (CQP). Dzięki nim, będziemy w stanie utworzyć relację zawierającą jedynie krotki spełniające wymagania doktora, np. aby pomiar temperatury wykonywany był przez certyfikowaną pielęgniarkę, z użyciem ustnego termometru. Odpowiednio, P = { *Oral(Instr), Certified(Patient, Date, Time), Valid(Value)* }.

Aby spełnić definicje, musimy wprowadzić najpierw predykat pomocniczy, *Temp(Patient,Date,Time,Nurse,Instr,Type)* który przygotowuje informacje o pomiarze temperatury, użytym instrumencie i imieniu pielęgniarki dokonującej pomiaru. *Temp* łączy się z każdym pomiarem, pielęgniarką i rodzajem termometru w zależności od czasu wykonania pomiaru.

Pierwsze dwa predykaty jakości (CQP) mogą zostać zdefiniowane następująco, z pomocą predykatu pomocniczego:

(6)

(7)

Pierwszy predykat jakości jest spełniony tylko wtedy, gdy użytym typem termometru jest termometr ustny (*tp*=oral). Drugi predykat może zostać wykorzystany do określenia czy pomiar został wykonany przez certyfikowaną pielęgniarkę.

Trzeci predykat zapobiega błędnemu uzupełnieniu wartości poprzez sprawdzanie czy wynik pomiaru mieści się w pewnym założonym możliwym przedziale wartości.

(8)

Z pomocą powyższych trzech predykatów, zgodnie z (2) możemy utworzyć nową relację:

(9)

Powyższa relacja powinna zawierać tylko pomiary spełniające wymagania doktora, co odpowiada instancji w Tabeli 2.

3.2 Jakościowa realizacja zapytań

Zapytania są napisane w języku powiązanym ze schematem S, skierowane do instancji D bazy danych. Dobrymi odpowiedziami na zapytania do D powinny być w zasadzie odpowiedzi na to same zapytanie zadane do instancji D’(I) lub D’p(I) . W konsekwencji oraz jako szczególny przypadek (1), dla zapytania , zbiorem *jakościowych odpowiedzi* na Q będzie:

(10)

Dla zapytań monotonicznych, np. złączeń, zawiera , gdzie ostatnie oznacza zbiór odpowiedzi na Q z D.

Odkąd R(D) są uzyskiwane jako zmaterializowane widoki kontekstowej instancji I, odpowiadanie na zapytania może zostać zrealizowane w oparciu o rozwijanie widoków, tzn. w celu uzyskania jakościowych odpowiedzi, określamy zapytanie które będzie kierowane do czystych relacji R’(I):

Algorytm Rozwijania Widoków: (QUA Quality Unfold Algorithm)

1. Zamień każdy predykat R w Q na odpowiadający mu R’ (lub R’P), uzyskując zapytanie Q’.
2. Zamień zapytanie Q’ na zapytanie , przez rozwijanie oparte na (2).
3. Jeśli trzeba, rozwiń definicje CQP, uzyskując „jakościowe zapytanie” , które może być zrealizowane na I.

Ostatni krok algorytmu otwiera możliwość rozważania CQP które nie są oparte tylko o schemat C i opierają się o zewnętrzne źródła, jak również reprezentujące inne predykaty jakości jak te przedstawione w [19].

Jakość D może być określona przez porównywanie każdego R(D), które wiąże się z R’p(I). Jest to tylko szczególny przypadek odpowiadania na zapytania: Zbiór czystych odpowiedzi na atomowe zapytania może być porównywany z odpowiadającymi R(D).

*Przykład 4.* Rozważmy zapytanie do schematu S naszego bieżącego przykładu, które pyta o temperaturę pacjenta, dnia 5 sierpnia (Sep/5): , które w notacji Datalog wygląda następująco:

(11)

W celu uzyskania jakościowych odpowiedzi, Q jest przepisywane pod względem schematu S’. Q jest trywialnie przepisywane jako: . Teraz TempNoon’P jest zdefiniowane przez (9). Przez rozwijanie widoku uzyskujemy:

(12)

To zapytanie może być wykonane bezpośrednio na I, które zawiera relację M, poprzez rozwijanie definicji (6)-(8) predykatów jakości lub bezpośrednie używanie ich rozszerzeń, jeśli zostały zmaterializowane.

Zauważmy, że w (12) mogliśmy mieć predykaty które nie są zdefiniowane tylko pod względem kontekstowego schematu C, ale również pod względem innych zewnętrznych źródeł. W konsekwencji zapytanie nie może być wykonane tylko na I, i może wymagać dodatkowych danych.

*Przykład 5.* (ciąg dalszy przykładu 4) Rozważmy teraz sytuację, w której zamiast posiadania instancji C(Nurse, Year), mamy jej definicję w znaczeniu zewnętrznego źródła #X(Nurse), które zawiera informacje certyfikowanych pielęgniarkach; #X(Nurse) zwraca *true* jeżeli podana pielęgniarka jest certyfikowana i *false* w przeciwnym przypadku. Odkąd C(Nurse, Year) jest częścią definicji predykatu CQP *Certified* (reguła (7)), realizacja nierozwiniętego zapytania w (12) spowoduje żądanie danych z #X(Nurse) w celu wartościowania predykatu *Certified.*

Niezależność predykatów jakości od kontekstowych danych czy źródeł jest szczególnie interesująca w przypadku, gdy chcemy użyć ich do filtrowania krotek z relacji, powiedzmy R w D. Ta sytuacja może być z łatwością uwzględniona w naszych ramach pracy. Dla predykatów , rozważamy kopię lub *alias,* . Każdy R’ posiada atrybuty i argumenty R, oraz ich dziedziny. Mamy również prostą definicję GAV dla , rozważając R jako *właściwe źródło,* w terminologii wirtualnej integracji danych [12] (popularne w definicjach widoków na pojedynczej instancji). Otrzymujemy kopię R’(D) z R(D) jako część kontekstowej instancji, a kontekstowa instancja staje się .

Teraz, gdy chcemy uzyskać jakościowe odpowiedzi na zapytanie Q skierowane do D (ze schematem S), biorąc pod uwagę predykaty jakości, zastępujemy każdy predykat in Q poprzez złączenie . Dane do realizacji dodatkowej formuły jakości ostaną pobrane z zewnętrznego źródła. Wynikowe zapytanie zostanie zrealizowane na I oraz rozszerzeniach dla predykatów jakości. Gdy brak tych drugich, realizacja zapytania może wywołać zapytania do zewnętrznych danych.

W tej części rozważamy wygodny, lecz niekoniecznie częsty przypadek, gdy instancja D, na której dokonujemy oszacowania jakości, jest kolekcją zmaterializowanych widoków instancji kontekstowej I. Alternatywne i naturalne przypadki, które musimy rozważyć mogą posiadać jedynie częściową kontekstową instancję I- wraz z metadanymi kontekstowych odniesień. Przestudiujemy to w części 4.

4. Brakujące kontekstowe dane

W przeciwieństwie do tego co mogło być zasugerowane w powyższych przykładach, nie możemy założyć, ze zawsze mamy dostęp do kontekstowej instancji I dla schematu C. Mogą to być niekompletne dane (puste) instancji I-, jak również dane z zewnętrznych źródeł. W tym bardziej ogólnym przypadku pojawia się sytuacja podobna do tych badanych w ramach systemów wirtualnej integracji danych. Tutaj, kontekstowy schemat zachowuje się jak pośredniczący, globalny schemat, a instancja D jako zmaterializowane źródło danych. W dalszej części będziemy się dokładniej zajmować tym połączeniem.

Załóżmy, że nie mamy kontekstowej instancji I dla schematu C, np. . Możemy zobaczyć D jako źródło danych dla systemu wirtualnej integracji danych , z globalnym schematem rozszerzającym kontekstowy schemat C [22, 4]. Możemy założyć że wszystkie dane w D są powiązane z przez C, jednak może zawierać więcej danych tego samego typu co te wniesione przez D. W konsekwencji, zakładamy że D jest otwartym źródłem dla . To założenie będzie ujęte poniżej w założonych i prawdziwych globalnych instancjach dla .

Jako że nie wszystkie dane w D muszą spełniać wymagania jakościowe w związku z C, potrzebujemy uwierzyć w relację pomiędzy D i jego spodziewaną jakościową wersją. W tym celu, podobnie jak w poprzednich przypadkach rozszerzamy C kopią S’ schematu Teraz i również staje się częścią globalnego schematu .

**Definicja 1.** Załóżmy że każde jest zdefiniowane jako widok Datalog: . (a) Prawdziwą instancją systemu jest instancja I- globalnego schematu, takiego jak: (a1) Dla każdego (a2) . (b) Instancją I schematu C jest prawdziwa kontekstowa instancja (LCI legal contextual instance) jeżeli istnieje I’ dla , taka że (restrykcje I’ dla schematu C).

Warunek (a1) szczególnie pociąga dane z D w kierunku . Prawdziwe instancje mają rozszerzenia danych z D gdy widoki definiujące predykaty R są obliczane. Wyrażenia w (a2) zachowują się jak globalne warunki integralności, np. na schemacie C’ i powodują efekt czyszczenia danych wirtualnie ładowanych do .

Możemy również rozważyć modyfikację tego przypadku gdzie w dodatku do D mamy jedynie fragment I- potencjalnej instancji kontekstowej. Oznacza to że mamy niekompletne dane kontekstowe. W tym przypadku Definicja 1 musi zostać zmodyfikowana poprzez dodanie warunku na I: (a3) , który wymaga aby prawdziwa instancja I’ była „zgodna” z częściową instancją I-. Przy , uzyskujemy poprzedni warunek. (przypis: Pojawienie się częściowej i zmaterializowanej globalnej instancji I- może być ujęte w scenariuszu wirtualnych systemów integracji danych, przez uznawanie I- jako oddzielnego właściwego źródła dla .)

**Definicja 2.** Krotka jest jakościową odpowiedzią na zapytanie jeżeli , gdzie Q’ jest uzyskane z Q poprzez zamianę każdego przez R’.

Jak poprzednio przyjmujemy jako zbiór jakościowych odpowiedzi na zapytanie Q do D przez C.

*Przykład 6.* Przyjrzyjmy się raz jeszcze zapytaniu Q(p, v) w (12) z Przykładu 4. Niech Q’(p,v) oznacza to samo zapytanie, teraz w związku ze schematem S’. Instancja D *TempNoon(Patient, Value, Time, Date)* w Tabeli 1 podlega ocenie jakości.

Teraz definiujemy wirtualny system integracji danych , z D jako otwartym źródłem, i relacjami w tabelach 3,4 i 5 tworzącymi częściową globalną instancję I-. W tym przypadku brakuje instancji (relacji) dla predykatu *M(Patient, Value, Time, Date, Instr)* w C.

Zgodnie z definicją 2, jakościowa odpowiedź na Q(p,v) musi zostać uzyskana z każdego LCI dla . Teraz każda LCI będzie zawierać krotki z *TempNoon(Patient, Value, Time, Date)* spełniające warunki stawiane w (9). W rzeczywistości tabela 2 odnosi się do najmniejszej LCI dla : Żaden jej podzbiór nie jest LCI ani żaden nadzbiór spełniający (9) również nie jest LCI. W konsekwencji pierwsza krotka tabeli 2 jest jedyną spełniającą dodatkowy warunek d = Sep/5. Otrzymujemy:

Odkąd zdefiniowaliśmy początkową instancję D jako otwarte źródło, możemy czerpać korzyści ze wszystkich algorytmów dla obliczania pewnych odpowiedzi na zapytania, które zakładają otwartość źródła [17]. Zakładając że te zapytania i definicje widoków są połączone, możemy wykorzystać np. *algorytm odwróconych zasad* [14]lub jego rozszerzenia [4, 11]. Zilustrujemy jego wykorzystanie w następującym przykładzie.

*Przykład 7.* (kontynuacja przykładu 6) Jeżeli odwrócimy definicję *TempNoon* w (5) otrzymamy:

(13)

Możemy zrealizować zapytanie w (12) poprzez rozwijanie definicji predykatu M, zgodnie z (13), uzyskując:

Przepisane zapytanie może być teraz zrealizowane na instancjach *TempNoon, S, C i T* (odpowiednio tabele 1, 3, 4, i 5). W ten sposób uzyskujemy taki sam wynik jak w poprzednim przykładzie.

Wreszcie, w związku z oceną jakości danych powstało kilka alternatyw. Jeżeli chcemy ocenić D, możemy rozważyć dla każdego LCI I instancję . Mamy . Możliwą miarą jakości może być . Analiza i porównanie tych dwu możliwych miar jakości zostanie dokonana w przyszłych pracach.

5. Powiązane prace

Badania nad jakością danych obejmują zarówno charakteryzację typów błędów, modelowanie procesów, w których mogą one powstawać jak również opracowywanie sposobów ich wykrywania oraz eliminacji. Większość podejść do powyższych problemów, bazuje na domniemanym założeniu, że błędy w danych pojawiają się szczególnie na poziomie symboliczno-syntaktycznym, tzn. rozbieżności pomiędzy wartościami (przykładowo Kelvin, a Kelvn).

Jak pokazano w [19], problemy z jakością danych pojawiają się także na poziomie semantycznym, tzn. jako rozbieżności pomiędzy nadawanymi im znaczeniami i interpretacjami. Dokładniej, nawiązując do [19], problemy takie mogą pojawić się w przypadku różnicy między zamierzonym znaczeniem (przypisanym przez producenta danych) i interpretacją ich znaczenia (dokonaną przez konsumenta danych). Taka rozbieżność często spowodowana jest niejednoznacznością w komunikacji między producentem, a konsumentem danych. Taka wieloznaczność jest nieunikniona, gdy zmienne czynniki (jak typ użytego termometru lub stan pacjenta) nie są wyraźnie odnotowane w danych (bądź metadanych). Oczywiście to, czy taka wieloznaczność jest uznawana za problem z jakością danych, zależy od celu w jakim dane te są używane.

W [19] przedstawiono zarys metod definiowania zarówno syntaktycznej, jak i semantycznej jakości danych w jednolity sposób, opartych na podstawowych pojęciach wartości i znaczeń. Zostało również przedstawionych wiele wysokopoziomowych predykatów jakości danych, opartych na porównywaniu symboli i znaczeń (ścisłe dopasowanie, częściowe lub jego brak). W niniejszej pracy podejmujemy kroki, mające na celu zaproponowanie konkretnego mechanizmu zdobywania i porównywania na poziomie semantycznym wymagań dotyczących jakości danych, korzystając z kontekstowych relacji i predykatów jakości oraz pokazania ich zastosowań w realizacji zapytań.

Użycie kontekstów w zarządzaniu danymi było proponowane już wcześniej (w [9] znajdują się badania), jednakże wykorzystywało inne sposoby zdobywania, reprezentowania oraz wykorzystania kontekstów. Przykładowo, kontekstowe informacje były wykorzystywane do wspierania półautomatycznych procesów projektowania widoków (opisane w [8]). Kontekst w [8] składa się z wielu elementów będących parami klucz-wartość (np. stanowisko=”przedstawiciel”, sytuacja=”na miejscu”, czas=”dziś”). Pewne graniczenia również mogą być przedstawiane w kontekście (przykładowo gdy stanowisko to „menadżer”, sytuacja nie może odbywać się „na miejscu”).

Określenie kontekstu pozwala na wybór małego widoku spośród dużej struktury bazy danych, który jest uznawany za trafny w tym kontekście. Projektowanie widoku będącego świadomym kontekstu, gdy kontekst jest doprecyzowany może odbywać się ręcznie lub półautomatycznie poprzez łączenie częściowych widoków zależnych od poszczególnych elementów kontekstu [10]. W niniejszej pracy kontekst jest określony w podobny sposób, jednak w innym celu niż w [8]. Głównym celem [8] jest redukcja rozmiarów (np.: wydzielenie użytecznych danych w danym kontekście), natomiast niniejsza praca jako główny cel stawia wybór danych w zależności od ich jakości (np.: znalezienie takiego podzbioru danych, który najlepiej spełnia pewne wymagania jakościowe).

Pracą najbardziej powiązaną z szacowaniem jakości danych jest [24]. Propozycja Naumann’a bazuje na uniwersalnej relacji [23] utworzonej z globalnego, relacyjnego schematu do integracji niezależnych źródeł danych. Zapytania są zbiorem atrybutów uniwersalnej relacji, z możliwymi warunkami nałożonymi na wartości atrybutów. Aby odwzorować zapytanie w źródłowych widokach, zapytania użytkownika są tłumaczone na zapytania do globalnego schematu bazy danych. Naumann wyróżnia kilka kryteriów jakości źródeł danych, takich jak ich wiarygodność, obiektywność, reputacja oraz weryfikowalność. Kryteria te, są wykorzystywane przy określaniu modelu jakości dla zapytań.

Zgodnie z [24], jakość zapytania jest określana następująco. Każde źródło otrzymuje punktację jakości informacji (IQ – information quality) dla każdego kryterium uznawanego za istotne. Punktacje te są następnie formowane w wektor, gdzie każdy element odpowiada innemu kryterium. Użytkownicy mogą wyrazić swoje preferencje dotyczące danego kryterium poprzez nadawanie wag poszczególnym elementom wektora, tworząc wektor ważony. Wektor ten jest wykorzystywany z kolei przez wieloatrybutowe metody podejmowania decyzji (MADM multi-attribute decision-making) w celu utworzenia rankingu źródeł danych wchodzących w skład uniwersalnej relacji. Metodami tymi może być zarówno proste skalowanie i sumowanie punktacji jak i złożone wzory oparte o macierze zgodności. Model jakości nie zależy od wybranych metod MADM, jeżeli korzysta z wag użytkownika i punktacji jakości IQ. Celem jest, biorąc pod uwagę wektory IQ źródeł, uzyskanie wektora IQ planu zapytania zawierającego źródła. Plan zapytania może być rozumiany jako drzewo połączeń między źródłami: liśćmi są źródła danych, natomiast pozostałymi węzłami są złączenia. Punktacje jakości informacji są obliczane dla każdego wewnętrznego węzła, idąc od liści ku korzeniowi. Ogólna jakość planu zapytania dana jest w postaci punktacji jakości korzenia tego drzewa.

Społeczność zajmująca się reprezentacją wiedzy (ang. knowledge representation community) wykonała pracę nad formalizacją i wykorzystaniem kontekstów. Postały ogólne idee w dziedzinie integracji i współpracy modeli i teorii. W [15] położono nacisk na właściwe interakcje lokalnych środowisk. W nowszych pracach, pojęcie kontekstu, czy też multi-kontekstu, zostało sformalizowane za pomocą *reguł mostowych* pomiędzy oznaczonymi kontekstami, gdzie każdy z nich może posiadać własną bazę wiedzy lub ontologię [12]. Reguły mostowe są wyrażane za pomocą rachunku zdań. Nie jest jasne czy potrafią one oddać bogate zależności znajdujące się w aplikacjach zarządzających danymi. Istnieje jeszcze niedawna praca o integracji rozproszonych logik opisowych [18], niekoniecznie odwołująca się do kontekstów, która pokazuje pomysły podobne do tych z literatury na temat kontekstów i reprezentacji wiedzy.

6. Podsumowanie

Zaproponowaliśmy ramy pracy pozwalające opisać bazę danych pod kątem jej jakości. Czynniki wpływające na jakość zostały ujęte jako interakcje instancji bazy danych z dodatkowymi, kontekstowymi źródłami danych bądź metadanych. W skład naszych ram pracy wchodzi mapowanie źródeł danych, które występuje na przykład w wirtualnej wymianie danych. Dobrej jakości odpowiedzi na zapytania, są zależne od współzależności pomiędzy danymi właściwymi, kontekstowymi i metadanymi.

Są to pierwsze kroki w kierunku wyrażania jakości danych i jakościowego realizowania zapytań jako czynności zależnych od kontekstu. Przeanalizowaliśmy kilka przypadków ogólnego postępowania, jak również dokonaliśmy kilku stwierdzeń na temat mapowania, widoków oraz zapytań. Ogólne ramy postępowania pozostają jednak wciąż do zbadania (więcej w [5, 6]). W tym celu powinno powstać i zostać zbadanych więcej algorytmów pozwalających na szacowanie jakości i dobrej jakości realizację zapytań.

Spośród wielu ciekawych obiektów prowadzonych badań, warto wspomnieć o: (a) Wykorzystaniu danych i zewnętrznych predykatów jakości w opisie instancji bazy danych. (b) Użyciu i integracji bardziej rzeczywistych i bezwzględnych predykatów jakości wprowadzonych w [19], w naszych ramach pracy. Mogą one obejmować tak szczegółowe aspekty jak składnia wartości, poprawność, sens, znaczenie etc. (c) Szczegółowej i porównawczej analizie miar jakości wspomnianych w tej pracy.

Konteksty pojawiły się już w literaturze dotyczącej zarządzania danymi, głównie w związku z oczywistymi aspektami danych jak np. geografia lub wymiary. Naszym zdaniem znacznie brakowało dotąd ogólnego pojęcia kontekstu, jego formalizacji i zastosowania w zarządzaniu danymi. Jest to najważniejszy problem, który musi zostać całościowo zbadany. Zaproponowaliśmy tylko kilka pierwszych pomysłów w tym kierunku.

Źródła

1. Abiteboul, S., Hull, R. and Vianu, V. Foundations of Databases. Addison-Wesley,

1995.

2. Ballou, D., Wang, R., Pazer, H. and Tayi, G. Modeling Information Manufacturing

Systems to Determine Information Product Quality. Management Science,

44(4):462-484, 1998.

3. Batini, C. and Scannapieco, M. Data Quality: Concepts, Methodologies and Tech-

niques. Springer, 2006.

4. Bertossi, L. and Bravo, L. Consistent Query Answers in Virtual Data Integration

Systems. In Inconsistency Tolerance, Springer LNCS 3300, 2004, pp. 42-83.

5. Bertossi, L. and Bravo, L. Query Answering in Peer-to-Peer Data Exchange Systems.

In ’Current Trends in Database Technology’, Springer LNCS 3268, 2004, pp.

478-485.

6. Bertossi, L. and Bravo, L. The Semantics of Consistency and Trust in Peer Data

Exchange Systems. Proc. International Conference on Logic for Programming, Ar-tificial Intelligence, and Reasoning (LPAR 07), 2007, Springer LNCS 4790, pp.

107-122.

7. Bleiholder, J. and Naumann, F. Data Fusion. ACM Computing Surveys, 41(1):1-41,

2008.

8. Bolchini, C., Curino, C., Orsi, G., Quintarelli, E., Rossato, R., Schreiber, F. and

Tanca, L. And What Can Context Do for Data? Communications of the ACM,

52(11):136-140, 2009.

9. Bolchini, C., Curino, C., Quintarelli, E., Schreiber, F. and Tanca, L. A Data-

Oriented Survey of Context Models. SIGMOD Record, 36(4):19-26, 2007.

10. Bolchini, C., Quintarelli, E. and Rossato, R. Relational Data Tailoring Through

View Composition. Proc. ER’07, Springer LNCS 4801, 2007, pp. 149-164.

11. Bravo, L. and Bertossi, L. Logic Programs for Consistently Querying Data Integration

Systems. Proc. International Joint Conference on Articial Intelligence(IJCAI'03), 2003, Morgan Kaufmann, pp. 10-15.

12. Brewka, G. and Eiter, Th. Equilibria in Heterogeneous Nonmonotonic Multi-

Context Systems. Proc. AAAI 2007, pp. 385-390.

13. De Giacomo, G., Lembo, D., Lenzerini, M. and Rosati, R. On Reconciling Data

Exchange, Data Integration, and Peer Data Management. Proc. PODS 2007. pp.

133-142.

14. Duschka, O., Genesereth, M. and Levy, A. Recursive Query Plans for Data Integration.

Journal of Logic Programming, 2000, 43(1):49-73.

15. Giunchiglia, F. and Serafini, L. Multilanguage Hierarchical Logics. Articial Intel-

ligence, 1994, 65:29-70.

16. Grahne, G. and Mendelzon, A. O. Tableau Techniques for Querying Information

Sources through Global Schemas. Proc. ICDT 1999. pp. 332-347.

17. Halevy, A. Answering Queries Using Views: A Survey. VLDB Journal, 2001,

10(4):270-294.

18. Homola, M. and Serafini, L. Towards Formal Comparison of Ontology Linking,

Mapping and Importing. In Proc. DL’10, CEUR-WS 573, 2010, pp. 291-302.

19. Jiang, L., Borgida, A. and Mylopoulos, J. Towards a Compositional Semantic

Account of Data Quality Attributes. Proc. ER’08, Springer LNCS 5231, 2008, pp.

55-68.

20. Kivinen, J. and Mannila, H. Approximate Inference of Functional Dependencies

from Relations. Theoretical Computer Science, 1995, 149:129-149.

21. Kolaitis, Ph. Schema Mappings, Data Exchange, and Metadata Management.

Proc. PODS’05, 2005, pp. 61-75.

22. Lenzerini, M. Data Integration: A Theoretical Perspective. Proc. PODS’02, 2002,

pp. 233-246.

23. Maier, D., Ullman, J. and Vardi, M. On the Foundations of the Universal Relation

Model. ACM Transactions on Database Systems, 1984, 9(2):283-308.

24. Naumann, F. Quality-Driven Query Answering for Integrated Information Systems.

Springer, 2002.

25. Wang, R. and Strong, D. Beyond Accuracy: What Data Quality Means to Data

Consumers. J. Management and Information Systems, 1996, 12(4):5-33.