Data Warehouse Architecture and Quality: Impact and Open Challenges

Matthias Jarke, Manfred A. Jeusfeld, Christoph J. Quix, Panos Vassiliadis, and Yannis Vassiliou

Abstract The CAiSE 98 paper "Architecture and Quality in Data Warehouses" and its expanded journal version [18] was the first to add a Zachman-like [37] explicit *conceptual enterprise modeling perspective* to the architecture of data warehouses. Until then, data warehouses were just seen as collections of – typically multidimensional and historized – materialized views on relational tables, without consideration of modeling of the (business) concepts underlying their structure. The paper pointed out that this additional conceptual perspective was not just necessary for a truly semantic data integration but also a prerequisite for bringing the then very active data warehouse movement together with another topic of quickly growing importance, that of data quality.

We were happy to see the citation and industrial uptake success of this paper as it played a central role in our European IST basic research project "Foundations of Data Warehouse Quality (DWQ)". Indeed, the paper was the first in a series of three CAiSE papers from 1998 to 2000 all three of which were selected as "best" CAiSE papers for expanded journal publication in *Information Systems* and

M. Jarke (✉) • C.J. Quix
Information Systems, RWTH Aachen University & Fraunhofer FIT, Ahornstr. 55, 52074 Aachen, Germany
e-mail: jarke|quix@cs.rwth-aachen.de

M.A. Jeusfeld
Information Management, Tilburg University, Tilburg, Netherlands
e-mail: manfred.jeusfeld@acm.org

P. Vassiliadis
Department Computer Science, University of Ioannina, Ioannina, Greece
e-mail: pvassil@cs.uoi,gr

Y. Vassiliou
DBLab, National Technical University of Athens, Athens, Greece
e-mail: yv@cs.ntua.gr

J. Bubenko et al. (eds.), *Seminal Contributions to Information Systems Engineering*,
DOI 10.1007/978-3-642-36926-1_14, © Springer-Verlag Berlin Heidelberg 2013

collected about 415 citations by end of 2012 according to Google Scholar. The final DWQ results were published in the book [19], still organized around basically the same architecture and quality model.

On a more personal note, it is worth mentioning that for the two junior co-authors (CQ, PV), this was their first major refereed publication, and has strongly influenced their follow-up research over more than a decade.

In this short note, we shall briefly summarize this own follow-up research as well as the impact on research and practice, in the three areas of data quality, data warehouse process engineering, and automated model management. We end with some ongoing research questions and open challenges.

1 Data Quality and Enterprise Integration

In 1998, the time was ripe for a serious treatment of quality as a first-class problem in information system engineering. Few years after the publication of the CAiSE'98 paper, both the necessity of handling data quality as a top-level concern and the idea of injecting quality properties in the metadata started gaining ground, as demonstrated by a proliferation of industrial efforts [2], books [3, 36], papers in top-ranked conferences and journals (e.g. [11]) and workshop series like DMDW, IQIS, and QDB. The CAiSE'98 paper contributed to the establishment of the idea that apart from relieving the operational systems from the query load, data warehouses also conceptually serve Inmon's "single version of the truth" principle for an organization.

A number of our own case studies confirmed this view and developed it further. In [30], we report the enormous impact of introducing DWQ-like semantic data cleaning and integration approaches into the worldwide financial reporting warehouse of Deutsche Bank, then one of the largest and most complex financial data warehouses worldwide. The project reduced the latency of consistent summary data from about 3 months to less than 1 day, at much better data quality. Subsequently, many business IT research groups expanded the conceptual modeling perspective from a management perspective [16], a user perspective [8], or the viewpoint of specific nonfunctional requirements [27].

In science and engineering applications, DW data often reflect project experiences, and our CAiSE'98 model had to be adapted for such knowledge warehouse settings. Already shortly after the CAiSE 98 paper, the Bayer company transferred our architectural concept to what they called their "process data warehouse" [20] for (chemical) process engineering. But this domain requires a richness of facets well beyond business applications, so it took our chemical engineering collaborators a decade to formulate an adequate, widely accepted set of core ontologies for this domain [7]. In a case study with Daimler, we also saw that data quality of long-lived data warehouses is often corrupted by creeping changes in the human

interpretation of the schemas, such that data mining techniques had to be developed to reverse-engineer the evolution of schema semantics over time [25]. Query processing over such multiple DW schema versions has been studied by [13].

Last not least, the quality models had to be made more efficiently usable. More than 100 KPI's from the literature were grouped into classes, with mappings to DW schemas. Moreover, it was noticed that quality metrics should not be kept separately but integrated directly into the architecture metamodel and its supporting repository. Manfred Jeusfeld extended ConceptBase, the system in which the CAiSE 98 models were first implemented, to include active rules and recursive functions with optimized execution by tabling prior function calls [17]. This enables natural definition of quality metrics even over hierarchically organized architectural and data elements. A similarly deep integration of quality into quality-aware DW reports has recently also been pursued at IBM [9].

2 Data Warehouse Process Engineering

With the benefit of the hindsight, an interesting omission of the CAiSE'98 paper was the treatment of software processes within a data warehouse. At the time the paper was authored, both the research and the industrial world viewed data warehouses from a static point of view. However, once the core problems of the design of the data architecture (and its contents) had been resolved, the main effort of data warehouse project teams has been devoted to the establishment of the refreshment process [23].

The CAiSE paper was the root of a research agenda that has lasted for more than a decade on the topic, technology, aiming at the establishment of ETL (Extract-Transform-Load) technology as a top-level topic in the data management and information systems engineering research communities [35]. Contributions have been made towards establishing methods that (a) allow administrators to design ETL workflows at conceptual and logical levels (e.g., [34]), (b) implement and tune these workflows at the physical level (e.g., [31]), and, (c) come up with efficient algorithms that can be incorporated in ETL tools to allow the efficient execution of ETL workflows (e.g., [28]). However, the first paper in this line of research came from practically the same team of authors of the CAiSE'98 paper, again in a CAiSE conference [33]. One can safely argue that the two papers should be considered as a pair as the CAiSE'98 paper covers the data architecture aspect and the CAiSE 2000 paper complements it with the management of operational processes for data warehouse metadata and quality.

Nowadays, both tasks are widely accepted in industrial practice – the ETL-based process perspective typically under the label of *Enterprise Application Integration*, the semantic data integration perspective under the label of *Enterprise Data Integration*. For both aspects, the OMG has in the meantime published some metamodel standards, such as the Common Warehouse Metamodel [29].

3 Automated Model Management

CWM also began to address another emerging issue, the growing heterogeneity of data models, by including source modeling packages not just for the relational model but also for XML or direct multidimensional models. But meanwhile, heterogeneity has gone much further. The explosion of IT in business and engineering (cyber-physical systems) has outpaced the possibilities of central data warehouses. Richer information integration architectures such as peer-to-peer networks, data stream management, or personal dataspaces are under investigation. The CAiSE'98 approach of carefully designing a central conceptual model as the basis for integration and quality is becoming infeasible, as a much higher degree of automation even in the handling of schemas/metamodels is required.

The first wave of this so-called *model management* movement [4] focused on introducing a *model algebra* with operators such as the automated generation of formal mappings by *matching* of schema elements, the semantically meaningful *merging* of schemas based on these mappings, and the *composition of mappings* as a basis for distributed query optimization, update propagation, or even schema evolution. In competition to programming solutions attempting to implement such an algebra, research on logic-based approaches continued.

In the end, it turned out that both approaches had to be combined. The key observation in the CLIO project at IBM Research was that the representation of mappings as simple correspondence links between schema elements are far too weak to allow for automated code generation and code optimization e.g. from composed mappings. These mappings needed to be expressed at least as (conjunctive) Datalog queries between any pair of sources to be integrate. For automated data integration, a new variant of so-called tuple-generating dependencies, *second-order tuple-generating dependencies* [12] were shown to allow correct and complete code generation even with composed mappings among relational sources.

In *model management 2.0* [5], model management is reconsidered under such richer mapping representations. In our work, we have aimed to extend the CLIO results to the case of *heterogeneous data models*: conceptual modeling formalisms such as UML or the ER model as well as the different kinds of structured and unstructured database models. A detailed analysis of the richness of these models, combined with the many subtle model variations in the chemical engineering case studies, led us to the conclusion that using the Telos language supporting by the ConceptBase system [26] would lead to a combinatorial explosion of subclass hierarchies which could not be handled with reasonable effort.

The GeRoMe metamodel [21] introduces a role concept at the metalevel which avoids this combinatorial explosion by using role annotations instead of subclassing, However, it maintains the efficient mapping of the conceptual modeling formalism to Datalog. In this way, we could show that query optimization and update propagation as in CLIO is possible even across an open architecture like a peer-to-peer network with heterogeneous data models among the peers [22]; in addition,

algorithms can be found to do schema merging in different scenarios not just with preservation of semantics, but also with minimization of the merged schemas [24].

4 Beyond Data Warehouses

In conclusion, we mention two further developments which at first glance seem much more revolutionary but surprisingly also show relationships to this work.

Firstly, we are observing a confluence of database, data warehouse, and search engine technologies. Naïve users expect to ask simple keyword questions also to structured databases, and conversely, many people want to ask structured queries a la SQL or multidimensional versions of it, to databases whose content is text or even multimedia objects. As one well-known example, the YAGO project extracts semantic knowledge in the form of RDF graphs from very large text bases such as Wikipedia [32]. Currently, this is being extended to a kind of RDF warehouse by adding temporal and spatial context [15]. Interestingly, a data quality framework for this web archiving similar to our CAiSE 98 approach has been recently developed [10].

The development of novel column-based main memory databases, such as SAP's HANA system, claims to void the need for separate data warehousing altogether [6, 14]. Other so-called NoSQL databases have also made broad claims, but each approach is typically best suited for particular applications and workload patterns, such that again, it is highly likely than an integration of multiple such non-standard database solutions with each other and with traditional databases will be necessary. At the operational level, a very nice approach to support such integration by a common programming framework has recently been proposed by [1] but it remains open what this implies for the enterprise architecture and for data quality management.

In summary, the field of architecture and quality in information integration appears alive and well for many years to come.

References

1. Atzeni P, Bugiotti B, Rossi L (2012) Uniform access to non-relational database systems. 24th Intl Conf Advanced Information Systems Engineering (CAiSE 2012), Gdansk/Poland, 160–174
2. Barateiro J, Galhardas H (2005) A survey of data quality tools. Datenbank-Spektrum 14: 15–21
3. Batini C, Scannapieco M (2006) Data Quality: Concepts, Methodologies & Techniques. Springer
4. Bernstein PA, Haas LM, Jarke M, Rahm E, Wiederhold G (2000) Is generic metadata management feasible? 26. Intl Conf Very Large Databases (VLDB 2000), Cairo/Egypt, 660–662

5. Bernstein PA, Melnik S (2007) Model management 2.0: manipulating richer mappings. ACM SIGMOD Conf., Beijing, China: 1–12
6. Bog A, Sachs S, Plattner H (2012) Interactive performance monitoring of a composite OLTP and OLAP workload. ACM SIGMOD Intl Conf Mgmt of Data, Scottsdale, Az, 645–648
7. Brandt SC, Morbach J, Miatidis M, Theißen M, Jarke M, Marquardt W (2008) An ontology-based approach to knowledge management in design processes. Computers & Chemical Engineering 32, 1–2: 320–342
8. Cappiello C, Francalanci C, Pernici B (2004) Data quality assessment from the user's perspective. ACM SIGMOD Workshop Information Quality in Information Systems, Paris, 68–73
9. Daniel F, Casati F, Palpanas T, Chayka O, Cappiello C (2008) Enabling better decisions through quality-aware reports. Intl Conf Information Quality (ICIQ), Cambridge/Mass
10. Denev D, Mazeika A, Spaniol M, Weikum G (2011) The SHARC framework for data quality in web archiving. VLDB Journal 20, 2: 183–207
11. Elmagarmid AK, Ipeirotis PG, Verykio VS (2007) Duplicate record detection: a survey. IEEE Trans. Knowl. & Data Eng. 19, 1: 1–16
12. Fagin R, Kolaitis P, Popa L, Tan WC (2005) Composing schema mappings: second-order dependencies to the rescue. ACM Trans. Database Systems 30, 4: 994–1055
13. Golfarelli M, Lechtenbörger J, Rizzi S, Vossen G (2006) Schema versioning in data warehouses: Enabling cross-version querying via schema augmentation. Data Knowl. Eng. 59, 2: 435–459
14. Grund M, Krüger J, Plattner P, Zeier A (2010) Cudré-Mauroux P, Samuel Madden S: HYRISE - A Main Memory Hybrid Storage Engine. PVLDB 4, 2: 105–116
15. Hoffart J, Suchanek FM, Berberich K, Weikum G (2013) YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. Artif. Intell. 194: 28–61
16. Holten R (2003) Specification of management views in information warehouse projects. Information Systems 28, 7: 709–751
17. Jeusfeld, M.A.; Quix, C.; Jarke, M. (2011) ConceptBase.cc User Manual Version 7.3. Technical Report, Tilburg University, http://arno.uvt.nl/show.cgi?fid=113912
18. Jarke M, Jeusfeld MA, Quix C, Vassiliadis P (1999) Architecture and quality in data warehouses: an extended repository approach. Inform. Systems 24, 3: 131–158.
19. Jarke M, Lenzerini M, Vassiliou Y, Vassiliadis P (2003) Fundamentals of Data Warehouses. 2nd edn., Springer.
20. Jarke M, List T, Köller J (2000) The challenge of process data warehousing. 26. Intl Conf Very Large Databases (VLDB 2000, Cairo/Egypt), 473–483.
21. Kensche D, Quix C, Chatti MA, Jarke M (2007) GeRoMe: a generic role-based metamodel for model management. J. Data Semantics 8: 82–117.
22. Kensche D, Quix C, Li X, Li Y, Jarke M (2009) Generic schema mappings for composition and query answering. Data & Knowledge Engineering 68, 7: 599–621
23. Kimball R, Caserta J (2004) The Data Warehouse ETL Toolkit. Willey
24. Li X, Quix C (2011) Merging relational views: a minimization approach. 30th Intl Conf Conceptual Modeling (ER 2011), Brussels/Belgium, 379–392
25. Lübbers D, Grimmer U, Jarke M (2003) Systematic development of data mining-based data quality tools. 26. Intl Conf Very Large Databases (VLDB 2003, Berlin/Germany), 548–559
26. Mylopoulos J, Borgida A, Jarke M, Koubarakis M (1990) Telos: representing knowledge about information systems. ACM Trans. Information Systems 8, 4: 325–362
27. Pardillo J, Trujillo J: Integrated model-driven development of goal-oriented data warehouses and data marts. 27th Intl Conf Conceptual Modeling (ER 2008), Barcelona, Spain: 426–439
28. Polyzotis N, Skiadopoulos S, Vassiliadis P, Simitsis A, Frantzell N-E (2007) Supporting streaming updates in an active data warehouse. 23rd Intl Conf Data Engineering (ICDE 2007), Constantinople, Turkey, 476–485
29. Poole J, Chang D, Tolbert D, Mellor D: *Common Warehouse Metamodel Developer's Guide*, Wiley Publishing, 2003

30. Schaefer E, Becker J-D, Boehmer A, Jarke M (2000) Controlling data warehouses with know-ledge networks. 26. Intl Conf Very Large Databases (VLDB 2000), Cairo/Egypt, 715–718
31. Simitsis A, Vassiliadis P, Sellis TK (2005) Optimizing ETL processes in data warehouses. 21st Intl Conf Data Engineering (ICDE 2005), Tokyo, Japan, 564–575
32. Suchanek FM,, Kasneci G, Weikum G (2007) Yago: a core of semantic knowledge. 16th Intl Conf World Wide Web (WWW 2007), Banff/Canada, 697–706
33. Vassiliadis P, Quix C, Vassiliou Y, Jarke M (2001) Data warehouse process management. Special Issue on Selected Papers from CAiSE 2000, Information Systems 26, 3: 205–236.
34. Vassiliadis P, Simitsis A, Georgantas PO, Terrovitis M (2003) A framework for the design of ETL scenarios. 15th CAiSE, Klagenfurt/Austria, 520–535
35. Vassiliadis P, Simitsis A (2009) Extraction-Transformation-Loading, In Liu L, Öszu T (eds.): Encyclopedia of Database Systems, Springer
36. Wang RY, Ziad M, Lee YW (2001) Data Quality. Advances in Database Systems 23, Kluwer
37. Zachman JA (1987) A framework for information systems architecture. IBM Systems Journal 26, 3: 276–292