Bus Inf Syst Eng 57(1):27–36 (2015)
DOI 10.1007/s12599-014-0365-x

RESEARCH PAPER

Capturing Enterprise Data Integration Challenges Using a Semiotic Data Quality Framework

John Krogstie

Received: 8 March 2014 / Accepted: 15 October 2014 / Published online: 3 February 2015

Abstract Enterprises have a large amount of data available, represented in different formats normally accessible for different specialists through different tools. Integrating existing data, also those from more informal sources, can have great business value when used together as discussed for instance in connection to big data. On the other hand, the level of integration and exploitation will depend both on the data quality of the sources to be integrated, and on how data quality of the different sources matches. Whereas data quality frameworks often consist of unstructured list of characteristics, here a framework is used which has been traditionally applied for enterprise and business model quality, with the data quality characteristics structured relative to semiotic levels, which makes it easier to compare aspects in order to find opportunities and challenges for data integration. A case study presenting the practical application of the framework illustrates the usefulness of the approach for this purpose. This approach reveals opportunities, but also challenges when trying to integrate data from different data sources typically used by people in different roles in an organization.

Keywords Enterprise data integration · Data integration · Data quality · SEQUAL

Accepted after two revisions by the editors of the special focus.

This paper is an extended version of the paper presented at PoEM 2013 (Krogstie 2013b).

Prof. J. Krogstie (✉)
Department of Computer and Information Science, Norges Teknisk-Naturvitenskapelige Universitet (NTNU), Sem Sælandsvei 7-9, 7491 Trondheim, Norway
e-mail: krogstie@idi.ntnu.no

1 Introduction

Fox and Gruninger (1998) describe an enterprise model as a computational representation of the structure, activities, processes, information, resources, people, behavior, goals, and constraints of a business, government agency, or other enterprise. Whereas we often think of enterprise models as visual models, the models are typically represented in some internal data/repository format, data formats that become important when we want to *integrate* enterprise knowledge from different tools supporting different roles in the organization. Data quality has for a long time been an established area of research (Batini and Scannapieco 2006) and work on quality assessment in data integration has also appeared as an area recently (Martin et al. 2012). A related area that was established in the 1990s is the quality of models (in particular the quality of conceptual data models) (Moody 1998). Data can be looked upon as a type of model (on the instance level) as illustrated, e.g., by the product models in a CAD systems. Traditionally, one has looked at model quality for models on the M1 (type) level (to use the model levels found in, e.g., MOF). On the other hand, it is clear especially in product and enterprise modeling that there are models on the instance level (M0), an area described as containing data (or objects in MOF terminology). Also if we look upon administrative data, e.g., upon persons, it is clear that this is an abstraction, focusing on certain properties (e.g., name, age) of people based on the purpose of obtaining the data, not of being a mirror of reality capturing all perceivable properties of someone. Thus, our outset is that also data quality can be looked upon relative to more generic frameworks for the quality of models. Integrating data sources is often incorrectly regarded as a primarily technical problem that can be solved by the IT-professionals themselves without involvement from the business side. This

widespread misconception focuses only on the data availability (e.g., in big data literature) as well as syntax and ignores the semantic, pragmatic, social, and purpose aspects of the data being integrated.

Discussions of data quality must be considered together with discussions of data model (or schema) quality. Comprehensive and generic frameworks for evaluating modeling approaches have been developed (Krogstie 2012; Nelson et al. 2012), but these are often too general for practical use. Inspired by the suggestion of (Moody 2005) for an inheritance hierarchy of quality frameworks, a specialization of the generic SEQUAL framework (Krogstie 2012) for the evaluation of the quality of data and their accompanying data models has been provided (Krogstie 2013a).

In Sect. 2, we present the problem area and case study for data integration. Whereas much work exists on technical solutions for data integration of similar data sources (Martin et al. 2012), the aim of this paper is rather to illustrate the variety of possible issues and problems for integration of enterprise data in practice, to be able to set a reasonable level of ambition in enterprise data integration efforts. The data quality framework (Krogstie 2013a) is described briefly in Sect. 3, followed by how it is used in the case in Sect. 4. In Sect. 5 we conclude, summarizing the experiences applying the SEQUAL specialization for the purpose of capturing (not solving) enterprise data integration challenges.

2 Description of the Problem Area of the Case Study

LinkedDesign[1] is an international project that aims to boost the productivity of engineers by providing an integrated, holistic view on data, actors, and processes across the full product lifecycle. To achieve this there is a need to evaluate the appropriateness of a selected number of existing data sources, to be used as a basis for the support of collaborative engineering in what is called a Virtual Obeya (Aasland and Blankenburg 2012). Obeya – Japanese for "large room" – is a term used in connection with project work in industry, where the focus is on the attempt to collect all relevant information from the different disciplines involved in the same physical room. A Virtual Obeya provides a virtual "room" with similar properties accessed by people taking different roles.

The selected data sources are of the types found particularly relevant in the use cases of the project. Relevant data sources for each industrial use case (in offshore engineering, automotive, and robot manufacturing) have been identified, in particular linked to user stories from the different cases. For a more concrete example, the case from offshore engineering relates to platform development using Knowledge-Based Engineering (KBE). KBE requires close collaboration between engineers in different domains, handling data in various data formats and a well structured knowledge acquisition technique followed by transparency and traceability in design automation. The KBE Design software is used by engineers with different roles in a product's lifecycle. The KBE Design goal is to visualize product knowledge and lifecycle information in an easily accessible way, enhance data integration and improve collaboration between the engineers involved. When we look at the *quality* of a data source (e.g., a PDM tool), we consider both the structure of the stored data (the data model, including meta-data) and the characteristics of the enterprise data itself, in light of our goal for reuse and revisualization of data, in a way that might result in annotated and/or updated data. The users perform collaborative work using the Virtual Obeya. The Obeya presents context specific information based on the people involved in the collaboration and other relevant information on products, projects, locations, tasks, tools, rules, guidelines, etc., i.e., the kind of information you typically find in enterprise models (Fox and Gruninger 1998). The data is mediated from existing work tools and is transformed depending on the context.

3 Introduction to a Framework for Data Quality Assessment

SEQUAL (Krogstie 2012) is a framework for assessing and understanding the quality of models and modeling languages in general. It has earlier been used for the evaluation of modeling and modeling languages from a large number of perspectives, including data (Krogstie 2013c), object-oriented, process (Recker et al. 2007), enterprise, and goal-oriented (Krogstie 2008) modeling. The specialization vs. data quality is rooted in the work of (Batini and Scannapieco 2006; Moody 1998; Price and Shanks 2005), and was originally presented in (Krogstie 2013a).

Looking at the sets of SEQUAL and data quality in the light of the cases of the LinkedDesign project, we find the following:

- ***G***: the goals of the modeling task. There are goals on two levels: the goal to be achieved when using the base tool, and the goal of supporting collaborative work using data from this tool as one of several sources of knowledge to be combined in the Virtual Obeya. Our focus in the case study is on this second goal.
- ***L***: The language is the way data is encoded (e.g., using a standard) and the language for describing the data model/meta-model.

[1] http://www.linkeddesign.eu.

- ***M***: The model, again on two levels, the data itself and the data-model.
- ***A***: Actors, i.e., the people in different roles using the models, with a specific focus on the collaborators in the use cases of the project.
- ***K***: The relevant explicit knowledge of the actors.
- ***I***: The social actor interpretation of the model. This relates to how easy it is for the different actors to interpret the data as it can be presented (in the base tool, and in a Virtual Obeya).
- ***T***: The tool interpretation of the model. Relates to the possibilities of the languages used to provide tool-support in handling the data (in the base tools, and in the Virtual Obeya)
- ***D***: Domain: The domain of modeling can on a general level be looked upon relative to perspectives captured in the generic EKA-Enterprise Knowledge Architecture of Active Knowledge Models (AKM) since these have shown to be useful for context-based presentation of enterprise knowledge integrated from different sources in other projects (Lillehagen and Krogstie 2008). Thus we look at enterprise information relative to: Products, tasks, goals and rules, roles (including organizational structure and persons, and their capabilities), and tools.

Based on this we can describe the data quality aspects more precisely for this case:

- *Physical quality:* The basic quality goal is that the externalized model ***M*** exists physically and is available to the relevant actors. In particular:
  - That the data is available in a physical format (and in different versions when relevant) so that it can be reused in the Virtual Obeya.
  - Possibility to store relevant meta-data, e.g., context information.
  - Availability of data for update or annotation/extension in the user interface.
  - Data is only available for those that should have access.
- *Empirical quality:* Empirical quality deals with comprehensibility of the data representation. This is not directly relevant when evaluating the data-sources per se. Guidelines for this are relevant when we look upon how data can be presented in tools (and in the Virtual Obeya).
- *Syntactic quality:* Syntactic quality is the correspondence between the model *M* and the language extension *L*. Are the data represented in a way following the defined syntax including standards for the area?
- *Semantic quality:* Semantic quality is the correspondence between the model *M* and the domain *D*. This includes both validity (often termed correctness) and completeness. Aspects of accuracy, consistency and precision are dealt with at this level (Krogstie 2013a). Do the data sources potentially contain the expected type of data and not other things? Note that we here look at the *possibility* of representing the relevant types of data, obviously the level of completeness is dependent on what is actually represented. Tools might also have mechanisms for supporting the rapid development of complete models.
- *Pragmatic quality:* Pragmatic quality is the correspondence between the model *M* and its actor interpretation (*I* and *T*). Is the data understood by the stakeholders?
- *Social quality:* Is there agreement among social actor's interpretations (*I*)? Since the data originates from different tools and often needs to be integrated in the Virtual Obeya, agreement on the interpretation of data and of the quality of the data sources among the involved stakeholders can be important. Aspects of reliability of the source are also important here.
- *Deontic quality:* The *deontic quality* of the model relates to that all statements in the model *M* contribute to fulfilling the goals of modeling *G,* and that all the goals of modeling *G* are addressed through the model *M*. Whereas the other levels relate to generic data quality aspects, we have here the possibility to address the particular goals of the cases explicitly. An important aspect of the case is to reduce waste in lean engineering processes (Manyika et al. 2009). In LinkedDesign, the use case partners and other project partners have prioritized the waste areas, and we have used this input to come up with the following list of waste to be avoided:
  - Searching: time spent searching for information.
  - Under-communication: Excessive or insufficient time spent with communication.
  - Misunderstanding: between different people, typically having different roles.
  - Interpreting: time spent on interpreting communication or artifacts.
  - Waiting: delays due to reviews, approvals, etc.
  - Extra processing: excessive creation of artifacts or information.

When we structure different aspects according to these levels, one will find that there might be conflicts between the levels (e.g., what is good for semantic quality might be bad for pragmatic quality and vice versa).

4 Evaluation of Relevant Tool Types

The research done is a case study primarily following an analytical approach, with a validation of results using

identified information within and outside the project on the different areas assessed. It is carried out

- by asking the project partners to reassess relevant input from previous project-deliverable to find the important tool types vs. overall requirements of the project and use cases in particular.
- by identifying the main tools of each type to evaluate based on what is used in the use cases.
- by adapting the SEQUAL approach for assessment of model quality as described in the previous section using this as an analytical lens for identifying possible issues.
- by evaluating the characteristics of the knowledge sources as for their applicability for being included and linked to the Virtual Obeya using the framework by a combination of literature search and interviews.
- by validating the result of different areas with domain and tool experts on the different tools and tool-types with people both inside and outside of the LinkedDesign project. The experts were first provided by our initial mapping of issue. We then went through each point, both to validate that our interpretation matched the expert interpretation and that issues where correctly positioned relative to the SEQUAL-framework. A few issues were removed since they were based on misunderstandings on our part. The discussion also introduced a few additional issues.

4.1 Evaluated Tools and Tool Types

In this project, based on the core needs of the use cases, we have focused on the following concrete tools and tool types in the assessment.

- *Office automation* much data relevant for engineers and other business professionals is developed and resides in office automation tools like Excel (Hermans 2012).
- *Computer-Aided Design (CAD)* is primarily used by the designers and some of the engineers, often in an early stage of product development.
- *Knowledge-Based Engineering (KBE)* has its roots in applying AI techniques (especially LISP-based) to engineering problems. In La Rocca (2012), four approaches/programming languages are described: IDL, GDL, AML, and Intent!, all being extensions of LISP. In LinkedDesign, one particular KBE tool is used: KBE design™. KBe Design™ is an engineering automation tool developed for Oil & Gas offshore platform engineering design and construction, built on top of a commercial KBE application (Technosofts AML). In the use case, there are two important data sources: The representation of the engineering artifacts themselves and the way the engineering rules are represented (in AML) as part of the code.
- *Product Lifecycle Management (PLM/PDM)* is the process of managing the entire lifecycle of a product from its conception through design and manufacture to service and disposal. Whereas CAD systems focus primarily on early phases of design, PLM attempts to take a full lifecycle view. PLM intends to integrate people, data, processes and business systems and provides a product information backbone for companies and their extended enterprise. There is typically a core group of people that creates information for such tools, and a vast group of people using this information.
- *Enterprise Resource Planning (ERP)* is defined as a "Framework for organizing, defining, and standardizing the business processes necessary to effectively plan and control an organization so the organization can use its internal knowledge to seek external advantage" (Blackstone 2008). We will in particular focus on SAP ERP that is in use in the organizations. ERP systems are very comprehensive, thus we will focus on the most relevant (seen from the use case partners) and core modules in SAP ERP. ERP systems have traditionally focused on internal process integration of well understood business functions, such as sales, production, and inventory management (Kelle and Akbulut 2005).

We here present an analysis of data quality relative to reusing enterprise data from all these tools, structured according to the semiotic levels of SEQUAL.

4.2 Quality Assessment

4.2.1 Features Supporting Physical Quality

Excel data: Data in tools like Excel can be saved both in internal formats, in open standards such as .html, .xps, .dif, and .csv as well as in open document formats (e.g., ods), and thus can be easily made available for visualization and further use. One can also export, e.g., PDF versions of spreadsheets for making the information available without a possibility for updates. Ensuring secure access to the data when exported can only be done manually. Since the format is known, it is possible to save (updated) data from a Virtual Obeya in the original spreadsheet.

CAD data: Data is stored in a local database and can be (partly) exchanged based on standard representations (see more under syntactic quality). Tools such as PDMS support simultaneous work of multiple users, with support for versioning and access control. In many tools, not only the product information is stored, but also the history-tree of operations needed for producing the model. It is also

possible to export data in generic formats (for visualization) like PDF.

KBE data: Rules and data are hard-coded in AML. One can export the AML-code into XML; however, some information might be lost in this process. There are also classes for querying the AML code, along with classes for automatic report creation. In KBE design it is in principle possible to interact with most systems. So far import/export routines to analysis software like GeniE and STAAD. Pro are implemented. Drawings can be exported to DWG (AutoCAD). When the model is held within the tool, access rights can be controlled, but it is hard to enforce this when the model is exported. There is limited support for controlling versions both in the rule-set and in the models developed based on the rule-set. As for the rules, these are part of the overall code which can be versioned. Some rules related to model hierarchy and metadata (not geometry) for export to CAD and PLM systems are stored in a database. A certain capability to import data contained in the CAD system PDMS is implemented.

PDM/PLM data: Core product data is held in an internal database supported by a common data model. The data can be under revision/version and access control. Some data related to the product might be held in external files, e.g., office documents. There can also be integration to CAD tools and ERP tools (both ways). In addition to accessing the data on a workstation, it is also possible to access the data on mobile platforms. Data can also be shared with, e.g., suppliers supporting secure data access across an extended enterprise.

ERP data: All the data in an ERP system is arranged in accordance to a central data dictionary that defines all the system's entities and their attributes and relationships. The data dictionary is based on the following structure:

- System configuration tables: Contain technical configuration data.
- Control tables: Contain parameters that govern the actual behavior of the system.
- Master data tables: Define the application data of the system.
- Transactional data tables: Individual purchase orders and sales orders are transactional data, and each document tends to be split up into different parts that are stored in different tables.

As with PLM systems it is possible to update the data of ERP systems from the outside and to export data to external systems. User profiles correspond to the daily tasks and responsibilities of the user and should ensure that a user can only perform transactions that belong to his job. Security/access control supported within the tool is hard to enforce when providing data external to the tool. SAP ERP transaction data can be exported to data warehouse tools using SAP Netweaver. The SAP HANA Architecture provides an in-memory database solution that can integrate transactional data and analytical queries.

4.2.2 Features Supporting Empirical Quality

Excel data: Excel has several mechanisms for data-visualizations in graphs and diagrams and these visualizations can be made available externally for other tools. The underlying rules and macros in the spreadsheets are not visualized.

CAD data: CAD tools have good functionality to visualize the product data in 3D. CAD has been a major driving force for research in computational geometry and computer graphics and thus for algorithms for visualizations.

KBE data: Geometric data can be visualized as one instantiation of a model with certain input parameters. There are also multiple classes for different kinds of finite element analysis on the model. Whereas the engineering artifact worked on is visualized in the work tool, the AML rules are not available for the engineer in a visual format. For those developing and maintaining the rule-base, the rules are represented as code (i.e., structured text).

PDM/PLM data: PLM tools typically support 2D and 3D visualization of the products within the tool. These are typically made in CAD tools, see above.

ERP data: The presentation of data within ERP system is normally achieved through traditional forms and tables. Tools to extract the overall process structure of the events and transaction (such as process mining tools) exist, which in this way can visualize not only the intended process, but also the process that is actually performed. Note that it is not particularly easy to do process mining towards SAP. van Giressel (2004) concludes that although SAP ERP logs all required data for process mining, it is not logged in a manner suitable for process mining. On the other hand successful approaches to solve the problem exist (Ingvaldsen 2011), but for this to be done automatically, the use of statistical techniques in addition to the log data and the reference model is necessary.

So-called reference models (also known as SAP's process ontology) exist to document all the standard functionality of an ERP system using EPC.

4.2.3 Features Supporting Syntactic Quality

Excel data: Although the syntax of the storage-formats for Excel is well-defined and standard data-types can be specified, there is no explicit information on the category of data (e.g., if the data represents product information). (Calculation) rules can be programmed, but these are undefined (in the formal meaning of the word).

CAD data: The international CAD data exchange standards include IGES and ISO 10303. ISO 10303 is informally known as STEP – STandard for Exchange of Product model data has so far been limited to the transfer of geometry information. Note that when exporting the model in STEP or IGES, the history-tree is not included. These standards have been incapable of handling parameters, constraints, design features, and other "design intent" data generated by modern CAD systems (Kim et al. 2007).

KBE data :In AML, data types are not defined. Programs might run even with syntax errors in formulas as there are both default values and other mechanisms in place to ensure that systems can run with blank values. The data is stored in a proprietary XML format, although it is also possible to make the model available using CAD standards, with only the information necessary for visualization available.

PDM/PLM data: Storage of PLM data is typically done according to existing standards. PLM XML is supported in the PLM tool Teamcenter, in addition to the formats needed for export to CAD and ERP tools mentioned under physical quality.

ERP data ERP data follows the rigorously defined data model, which also can be accessible from the outside, although the data dictionary is not adaptable to external data models. The data is implemented in relational databases.

4.2.4 Features Supporting Semantic Quality

Excel data: You can represent knowledge of all the listed categories in a spreadsheet, but since the data model is implicit, it is not possible to know what kind of data you have available without support from the human developer of the data, or by having this represented in some other way.

CAD data: CAD systems typically focus on the (geometric) representations of products at the instance level. One might also represent some rules relative to the product in CAD systems, but one cannot capture knowledge on organizational structure, tools, and underlying business processes in these tools. Another limitation to most CAD tools is lacking representation of the functions (e.g., overall goals) of the different parts of the design, although some tools support the representation of (functional and non-functional) requirements.

CAD tools usually support the development of element catalogues. The catalogue contains standard reference data for the available types of components. This can support rapid development of new structures, i.e., rapid completion of the product model. The support of default values in the tool user interface can also be useful in this respect. CAD tools are often integrated with different analysis tools which can support the development of valid design models.

KBE data: The focus in KBE Design is the representation of product data. AML is used to represent engineering rules related to product design. There are also possibilities to represent process information related to the products. Note that an OO-framework has some limitations to representing rules, e.g., for representing rules spanning many classes. The AML framework also supports dependency tracking, which means that if a value or rule is updated, everything that uses that value or rule is also changed. Dynamic instantiation is supported, providing a potential short turnaround for changes to the rule-set.

PDM/PLM data As the name implies, the main data kept in PLM systems is product data, including data relevant for the product lifecycle process. Schedule information and workflow modeling is supported in tools such as Teamcenter, but similar to CAD tools, the function of the parts in the product is not represented in most tools. Compliance management modules can support representation of regulations (as a kind of rules).

ERP data: ERP systems have a strong emphasis on process information. Also information of products and organizational structure and roles is usually captured. Business rules and policies are usually not explicitly stored in an ERP system, even though they are reflected in the configuration of system components. As for tool information (e.g., what tool is used in which steps), this is typically not represented conceptually. Completeness of data is often enforced in the tool, whereas validity is more difficult to enforce. On the other hand, since modules share a common data dictionary, an ERP system can to a large extent verify the consistency of data across modules and business areas.

4.2.5 Features Supporting Pragmatic Quality

Excel data: As indicated under empirical quality you can visually present data in spreadsheets which can be shared (and you can potentially update the visualization directly), but as discussed under semantic quality, no explicit knowledge of the type of the data represented exists.

CAD data: CAD tools support numerous ways of visualizing the design in an integrated manner, e.g., 3D-view, product model trees, etc. Viewing mechanisms showing only parts of the overall structure through layering (e.g., the parts relevant for one discipline), relating this to the overall structure is a very important mechanism to be able to handle the complexity of CAD models. Another interesting approach for ensuring comprehension is the export of the CAD data to prototyping tools such as 3D-printers. While the goal of CAD systems is to increase efficiency, they are not necessarily the best way to allow newcomers to understand the geometrical principles of product modeling. For this, scripting languages have been developed.

The scripting approach is preferred by some CAD instructors as scripts reveal all details of the design procedure.

KBE data: The experiences from the use case indicate that it is very important to be able to provide rule visualizations, and that these can be annotated with meta-data and additional information. Standard classes allow you to query AML models, generating reports. Data can be visualized any way you want in AML, and if the required visualization is not part of the standard AML framework, then it can be created. It is practical to have everything working in the same environment, but it can be difficult for non-experienced users to find the right functionality.

PDM/PLM data: Relevant context information can be added to the product description supporting understanding. PLM systems have become very complex and as such more difficult to use and comprehend. The size of the products (number of parts) has also increased over the years. Reporting is traditionally in Excel, but newer tools can provide reports as annotations to 3D-models.

ERP data: The majority of research on ERP systems has been on the phases of configuration, implementation and deployment. More focus on better use and continuous improvements of the systems in use is needed. As described under empirical quality, the user interface of ERP systems is in traditional forms and tables, although also a hierarchical structure for accessing relevant transactions is provided. The concept of master data is behind the pragmatic quality support given by SAP ERP. Master data serves two purposes: (1) Consistency across transactions, (2). Ease of data entry. Because all transaction data in ERP systems are linked to one or more master data table, additional information is automatically filled out. Therefore, this is a very economical way of entering, expressing, retrieving and manipulating large data sets.

There are more than 10 000 transaction codes, thus some structuring is necessary here. Decision making often involves looking at several different tables to obtain a complete overview of a situation. To support decision making there is a need for graphical information that goes beyond single tables or single graphs to get a broader picture of the system (Parush et al. 2007). An experimental study by Parush et al. (2007) found that with users of ERP systems for supply chain management, graphical visualization of data improved performance, especially for inexperienced users. Visualization enables professionals to comprehend information quickly, and allows immediate action (Chorafas 2005). With the lack of visualization in ERP systems, juxtaposed with the recognized benefits of useful data display, visualization emerges as a requirement for improved ERP functionality. The visual process models in EPC (the reference process model and the adapted models) are not available for the normal end-user.

4.2.6 Features Supporting Social Quality

Excel data: Since Excel is a personal tools (often adapted to personal needs, even in cases where a company-wide template has been the starting point), there is a large risk that there are inconsistencies between data (and the underlying data model) in different spreadsheets and between data found in spreadsheets and in other tools.

CAD data: CAD is primarily used by the designers and some of the engineers, primarily in an early stage of product development. Thus, the agreement on these data might be less than, e.g., data developed in more organizationally integrating tools such as PLM and ERP tools. On the other hand, the part of the data that is stored using established standards would probably be easier to ensure if interpreted identically across user-groups and organizations, with the limitation that not all the relevant information is captured in these interchange formats. STEP is developed and maintained by the ISO technical committee TC 184, *Automation systems and integration*, sub-committee SC 4, *Industrial data*. Like other ISO and IEC standards, STEP is copyright by ISO and is not freely available. However, the ISO 10303 EXPRESS schemas are freely available, as are the recommended practices for implementers. Also, since data in CAD tools represents physical products, it is probably easier to agree on this than on more conceptual data.

KBE data: KBE is a particular solution for engineering knowledge, and experiences from the use case indicate that there is not always agreement on the rules represented. The KBE Design tool is used for developing oil-platform designs, but for other engineering and design tasks other tools are used. Export to tools used company-wide such as PDMS is important to establish social quality of the models.

PDM/PLM data: PLM systems are systems for integrating the enterprise. When implementing PLM systems one needs to agree on the system set-up, data-coding, etc. across the organization. Thus, when these kinds of systems are successfully implemented, one can expect high agreement on the data found in the tool across the organization. The so-called work and benefit disparity might occur [this problem was originally described in connection with groupware systems (Grudin 1994)]. Company-wide applications often require additional work from individuals who do not perceive a direct benefit from the use of the application. When, e.g., creating new parts, a large number of attributes needs to be added, thus it takes longer time to enter product-information in the beginning.

ERP data: ERP-solutions such as SAP ERP are widely used. Since ERP systems are meant to integrate data from many sources, the organizational agreement of the data found in these will often be high (at least when the tool has

been in operation successfully for a while). Note that the integrated nature of ERP tools means that one in many cases might need to enter data not only useful for one's own task, but also data mainly useful for others.

4.2.7 Features Supporting Deontic Quality

We summarize the main points here relative to factors for waste reduction in lean engineering.

Excel Data:

- Searching: when Excel is used, often data exist in a number of different Excel-sheets developed by different people, and it can be hard to know if you have the right version.
- Under-communication: there is no explicit data-model, thus the interpretation of data might be based on labels only, which can be interpreted differently by different people. A number of (calculation) rules are captured in Excel-sheets without being visualized.
- Misunderstanding: due to potentially different interpretations of terms, misunderstandings are likely.
- Interpreting: since the meaning of data is under-communicated, the time for interpretation might be long.
- Waiting: if data must be manually transformed to another format to be usable, this might be an issue.
- Extra processing: due to the versatility of tools like Excel, it is very easy to represent additional data and rules, even if they are not deemed useful by the organization.

CAD Data:

- Searching: finding the relevant CAD data can be made easier by linking it to enterprise tools such as PLM tools.
- Under-communication: in CAD tools, there might be limitations to the representation of underlying design rules and process information.
- Misunderstanding: due to the number of assumptions that are included in a product design that is not represented explicitly, there might be misunderstandings at a later stage.
- Interpreting: a number of tools exist to carry out different types of analysis, which might make it easier to support interpretation of the product model.
- Waiting: if others than designers and engineers need information from CAD tools, or need to have changes done at a later stage, they might be dependent on the availability of the engineer to conduct the changes.
- Extra processing: since CAD tools store the geometry on the instance level, reuse for, e.g., variants of products might take extra time.

KBE Data:

- Searching: representing all rules in the KBE system is useful, but they are only structured to a limited degree, e.g., relative to how rules influence each other.
- Under-communication: since AML-rules are accessible as code only, it can be hard to understand how different design decisions are enforced.
- Misunderstanding: these can result from not having access to the rules directly.
- Interpreting: additional time might be needed for interpretation for the above mentioned reason.
- Waiting: if you do not receive support quickly for updating rules (if necessary), this can be an issue. The use of dynamic instantiation described under semantic quality can alleviate this, on the other hand people with specific skills are required to add or change rules.
- Extra processing: there may be a need to represent rules differently to be useful in new situations. On the other hand, this can be addressed by using the abstraction mechanisms well.

PDM/PLM Data:

- Searching: large models and a lot of extra data might make it difficult to obtain an overview and find all the relevant information. On the other hand, since a common data model exists, it should be easier to find all data relevant for a given product.
- Under-communication: since extra data has to be added up front for the use later in the product life cycle, there is a danger that not all necessary data is added (or is added with poor quality), which can lead to the next two issues:
- Misunderstanding: this can be a result of under-communication.
- Interpreting: when engineers and other groups need to communicate, one should also be aware of possible misunderstandings, given that it seems to be hard to learn these tools if you are not an engineer. Also given that only a few people actually add data, a lot of people need to interpret this data without actively producing it.
- Waiting: it can be a challenge when a change has been made for this to be propagated to, e.g., ERP systems and supplier systems. For some type of data this propagation can be made automatically.
- Extra processing: as it is necessary to add data up front, this can be a challenge when you need to perform changes, to have the data produced in earlier phases updated.

ERP Data:

- Searching: a trend of enterprise systems is that they cover more and more business functions, connecting back-office operations with front-office systems, as

well as moving towards real-time tracking and monitoring of operations (Lyons 2005). The result is that the amount of information within ERP systems is increasing. This means improved possibilities for decision making, but also poses a challenge as to how to present data in a way that is useful for those making the decisions. On the other hand, since a common data model exists, it is potentially easier to find relevant data. ERP systems are known for poor search interfaces. Generally, ERP systems are good for precise queries, but less good for vague exploratory queries.

- Under-communication: a possible effect of the work-benefit disparity issue is as with PLM systems that some data might not be available in good quality.
- Misunderstanding: this can be a result of under-communication.
- Interpreting: the limited support for pragmatic quality might make the interpretation of data difficult.
- Waiting: if one depends on others filling in relevant data, this can result in unnecessary waiting.
- Extra processing: if data has been provided, but this is inaccurate, extra processing is necessary to correct it.

We summarize the main points from the discussion in the Table 1 below, with a focus on issues that might make integrated usage of data from different sources specifically problematic. To identify these kinds of issues it is useful to look at differences between the different sources at the same quality level (same row). We will return to some main points in the next section.

5 Summary and Conclusion

Above, we have put together five assessments carried out using the specialization of SEQUAL for data quality. Overall, the evaluation indicates opportunities, but also challenges when trying to integrate data from different knowledge sources typically used by people in different roles in an organization, and presenting this data in a common user interface. In particular, it highlights how

Table 1 Overview of main issues for data integration

	Excel	CAD	KBE	PDM/PLM	ERP
Main users	General business professionals at all development stages	Designers and engineers at early development stages	Engineers, early development	Potentially all working with a product at some stage in the lifecycle	All involved in the business process, all stages
Physical	(Unsecure) files in file system in standard format. Data-loss in export	Internal database. Data-loss in export	Rules coded in AML. Data-loss in export	Internal database with links to external files. Data-loss in export	Database governed by central data dictionary. Data-loss in export
Empirical	Flat table representation with visualization possibility for some data	2D and 3D visualization of product data	Individual rules. No rule-structure-visualization	2D and 3D visualization of product data	Database tables, process knowledge can be extracted and visualized as process models
Syntactic	Data can represent anything, data-type adherence only	Support exchange standards (e.g., STEP)	Programs can run with syntactic errors. Support exchange standards	Support exchange standards	Adherence to central data model
Semantic	No explicit semantics	Product data. Some rules can be represented	Engineering rules and product data. Some process information	Product data, product lifecycle (process), some rules	Process and organizational information
Pragmatic	Limited support	Many visualizations and viewing mechanisms available.	Rule visualization is lacking. Report generation	Representation of context information. Reporting in Excel and around 3D-models	Master data to ensure consistency. Limited visualization
Social	Personal tools, no pre-existing organizational agreement on data	Used by specialist, can support common interpretation through use of standards	Frequently disagreement on rules between developers and engineers	Enterprise-integrating tool. Work-benefit disparity issue	Enterprise-integrating tool. Work-benefit disparity issue
Deontic	Potential issues on all categories	Under-communication of function. Work depending on experts	Under-communication, depending on code-format	Comprehensive models, but can be hard to find what is relevant and interpret it.	Large amount of information, can be hard to find all relevant information

different tools have a varying degree of explicit meta-model (data model), and how this is externally available in varying degrees. For example, in many export formats one loses some of the important information on product data. Even when different tools support, e.g., process data, these are often process data of different granularity which might be difficult to integrate automatically. The tools themselves all possess challenges relative to the categories of waste in lean engineering. In a Virtual Obeya one would want to combine data from different sources based on the context to address these reasons for waste. Somewhat dependent on the concrete knowledge sources to combine, this indicates that it is often a partly manual (and large) job to prepare for such combinations. Also the different levels of agreement of data from different sources (social quality) will influence the use of schema and object matching techniques in practice.

As described in the introduction, in this paper we have mainly illustrated potential issues when attempting data integration in an organizational setting. A limitation of the work is that since it is a case study, we might not have identified all potential issues. On the other hand, in many cases the data to be integrated comes from a more homogenous set of sources, making the problem less complex, thus the generalizability of the results can, as in any case study, be challenged. Future work on the data quality framework will include to devise more concrete guidelines and metrics and evaluate the adaptation and use of these empirically in other cases, especially concerning how to make trade-offs between the different data quality types. Furthermore we will look at newer work (Batini et al. 2009; Jiang et al. 2009) in the area in addition to what we have mapped so far. This will also make it easier to integrate our work with existing work on data integration quality assessments (Martin et al. 2012). Finally we will look more upon the use of the framework when integrating data from less technical areas such as CRM systems and data from social networks, and issues in the area of big data.

Acknowledgments The research leading to these results was done in the LinkedDesign project that has received funding from the European Union Seventh Framework Programme ([FP7/2007-2013]) under grant agreement n 284613.

References

Aasland K, Blankenburg D (2012) An analysis of the uses and properties of the Obeya. In: Proceedings of the 18th International ICE-Conference, Munich, 2012

Batini C, Scannapieco M (2006) Data quality: concepts, methodologies and techniques. Springer, Berlin

Batini C, Cappiello C, Francalanci C, Maurino A (2009) Methodologies for data quality assessment and improvement. ACM Comput Surv (CSUR) 41(3), article 16

Blackstone JH (2008) APICS dictionary. APICS, Athens

Chorafas DN (2005) The real-time enterprise. Auerbach, Boca Raton

Fox MS, Gruninger M (1998) Enterprise modeling. AI Mag 19(3):109–121

Grudin J (1994) Groupware and social dynamics: eight challenges for developers. Commun ACM 37(1):92–105

Hermans FFJ (2012) Analyzing and visualizing spreadsheets. PhD thesis, Software Engineering Research Group, Delft University of Technology, The Netherlands

Ingvaldsen JE (2011) Semantic process mining of enterprise transaction data. PhD thesis, NTNU, Trondheim

Jiang L, Barone D, Borgida A, Mylopoulos J (2009) Measuring and comparing effectiveness of data quality techniques. In: Proccedings of CAiSE 2009, pp 171–185

Kelle P, Akbulut A (2005) The role of ERP tools in supply chain information sharing, cooperation, and cost optimization. Int J Prod Econ 93–94:41–52

Kim J, Pratt MJ, Iyer R, Sriram R (2007) Data exchange of parametric cad models using ISO 10303–108, NISTIR 7433

Krogstie J (2008) Integrated goal, data and process modeling. In: Johannesson P, Søderstrøm E (eds) Information systems engineering from data analysis to process networks. IGI, Pennsylvania, pp 43–65

Krogstie J (2012) Model-based development and evolution of information systems: a quality approach. Springer, Berlin

Krogstie J (2013a) A semiotic framework for data quality. In: Proceedings of EMMSAD 2013, Valencia, pp 395–410

Krogstie J (2013b) Evaluating data quality for integration of data sources. In: Proceedings of PoEM 2013, Riga, pp 39–53

Krogstie J (2013c) Quality of conceptual data models. In: Proceedings of ICISO 2013, Stockholm, pp 168–176

La Rocca G (2012) Knowledge based engineering: between AI and CAD. Review of a language based technology to support engineering design. Adv Eng Inform 26(2):159–179

Lillehagen F, Krogstie J (2008) Active knowledge modeling of enterprises. Springer, Berlin

Lyons MH (2005) Future ICT systems – understanding the business drivers. BT Technol J 23(5):11–23

Manyika J, Sprague K, Yee L (2009) Using technology to improve workforce collaboration: what matters. McKinsey Digital

Martin N, Poulovassillis A, Wang J (2012) A methodology and architecture embedding quality assessment in data integration. ACM J Data Inf Qual 4(4), article 17

Moody DL (1998) Metrics for evaluating the quality of entity relationship models. In: Proceedings of the seventeenth international conference on conceptual modelling (ER '98), Singapore. Lecture Notes in Computer Science 16–19, pp 211–225

Moody DL (2005) Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. Data Knowl Eng 55:243–276

Nelson HJ, Poels G, Genero M, Piattini M (2012) A conceptual modeling quality framework. Softw Qual J 20:201–228

Parush A, Hod A, Shtub A (2007) Impact of visualization type and contextual factors on performance with enterprise resource planning systems. Comput Ind Eng 52(1):133–142

Price R, Shanks G (2005) A semiotic information quality framework: development and comparative analysis. J Inf Technol 20(2):88–102

Recker J, Rosemann M, Krogstie J (2007) Ontology- versus pattern-based evaluation of process modeling language: a comparison. Commun Assoc Inf Syst 20:774–799

van Giressel N (2004) Process mining in SAP R/3. Master thesis, Eindhoven University of Technology, The Netherlands