Quality Measures for ETL Processes

Vasileios Theodorou[1], Alberto Abelló[1], and Wolfgang Lehner[2]

[1] Universitat Politècnica de Catalunya, Barcelona, Spain
{vasileios,aabello}@essi.upc.edu
[2] Technische Universität Dresden, Dresden, Germany
wolfgang.lehner@tu-dresden.de

Abstract. ETL processes play an increasingly important role for the support of modern business operations. These business processes are centred around artifacts with high variability and diverse lifecycles, which correspond to key business entities. The apparent complexity of these activities has been examined through the prism of Business Process Management, mainly focusing on functional requirements and performance optimization. However, the quality dimension has not yet been thoroughly investigated and there is a need for a more human-centric approach to bring them closer to business-users requirements. In this paper we take a first step towards this direction by defining a sound model for ETL process quality characteristics and quantitative measures for each characteristic, based on existing literature. Our model shows dependencies among quality characteristics and can provide the basis for subsequent analysis using Goal Modeling techniques.

Keywords: ETL, business process, quality measures.

1 Introduction

Business Intelligence nowadays involves identifying, extracting, and analysing large amount of business data coming from diverse, distributed sources. In order to facilitate decision-makers, complex IT-systems are assigned with the task of integrating heterogeneous data deriving from operational activities and loading of the processed data to data warehouses, in a process known as Extraction Transformation Loading (ETL). This integration requires the execution of real-time, automated, data-centric business processes in a variety of workflow-based tasks. The main challenge is how to turn the integration process design, which has been traditionally predefined for periodic off-line mode execution, into a dynamic, continuous operation that can sufficiently meet end-user needs.

During the past years, there has been considerable research regarding the optimization of ETL flows in terms of functionality and performance [26, 7]. Moreover, in an attempt to manage the complexity of ETL processes on a conceptual level that reflects organizational operations, tools and models from the area of Business Process Management (BPM) have been proposed [29, 3]. However, the dimension of process quality [25] has not yet been adequately examined in a systematic manner. Unlike other business processes, important quality factors for ETL process design are tightly coupled to information quality while

L. Bellatreche and M.K. Mohania (Eds.): DaWaK 2014, LNCS 8646, pp. 9–22, 2014.

depending on the interoperability of distributed engines. Added to that, there is increasing need for process automation in order to become more cost-effective [28] and therefore there needs to be a common ground between business users and IT that would allow the first to express quality concerns in a high level language, which would automatically be translated to design choices.

In this paper we take a first step towards quality-aware ETL process design automation by *defining a set of ETL process quality characteristics and the relationships between them, as well as by providing quantitative measures for each characteristic*. For this purpose, we conduct a systematic literature review, extract the relevant quality aspects that have been proposed in literature and adapt them for our case. Subsequently, we produce a model that represents ETL process quality characteristics and the dependencies among them. In addition, we gather from existing literature metrics for monitoring all of these characteristics and quantitatively evaluating ETL processes. Our model can provide the basis for subsequent analysis that will use Goal Modeling techniques [19] to reason and make design decisions for specific use cases, using as input only the user-defined importance of each quality characteristic.

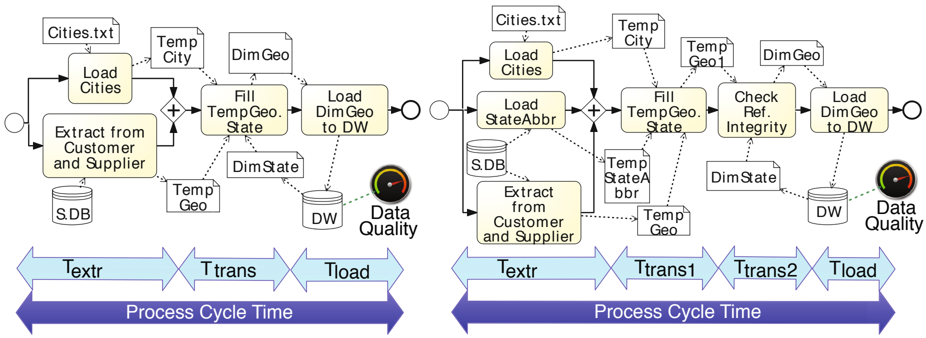

Fig. 1. Example alternative ETL process design with same functionality

We illustrate how our model works through a running example, borrowing the use case from the work of El Akkaoui and Zimanyi [11] . The use case is an ETL process that extracts geographical data from a source database (S.DB) and after processing, loads a dimension table into a data warehouse (DW). The tables extracted from the source database are *Customer* and *Supplier*, which contain information about *City*, *State*, *Country* and *ZipCode*. However, the attribute *State* might be missing from some records and therefore a flat text file (Cities.txt) containing *City*, *State*, and *Country* tuples is also used. After *State* entries have been filled, the table *DimGeo* is loaded to the data warehouse, with attributes *City*, *StateKey* and *ZipCode*, where the *StateKey* for each state is derived from *DimState*, which is another dimension table in the data warehouse. This process is modelled using the Business Process Model and Notation (BPMN[1]) and two alternative designs can be seen in Fig. 1.

[1] `http://www.bpmn.org`

The paper is organized as follows. Section 2 presents related work regarding quality characteristics for design evaluation. In Section 3 we present the extraction of our model from related work. The definitions, measures and dependencies among characteristics are presented in Section 4 and Section 5 where we distinguish between characteristics with construct implications and those only for design evaluation, respectively. Finally, we provide our conclusions and future work in Section 6.

2 Related Work

The significance of quality characteristics for the design and evaluation of ETL processes has recently gained attention. Simitsis et al. [28] recognise the importance of considering not only process functionality but also quality metrics throughout a systematic ETL process design. Thus, they define a set of quality characteristics specific to ETL processes that they refer to as *QoX metrics* and provide guidelines for reasoning about the degree of their satisfaction over alternative designs and the tradeoffs among them. A more recent work that has also considered the ideas from [28] is the work from Pavlov [24]. Based on well-known standards for software quality, the author maps software quality attributes to ETL specific parameters which he calls *QoX factors*. He defines these factors in ETL context and reasons about the impact that the different ETL subsystems might have on each characteristic.

Focusing on Information Quality, Naumann [23] provides a comprehensive list of criteria for the evaluation of Information Systems for data integration. In the same area, Dustdar et al. [10] identify most important challenges for Data Integration and highlight quality concerns in distributed, heterogeneous environments. Likewise, Jarke et al. [15] identify the various stakeholders in Data Warehousing activities and the differences in their roles as well as the importance of reasoning among alternative quality concerns and how that affects design choices.

In the last years, there has been an effort in the area of Business Process Management to quantify process quality characteristics and to empirically validate the use of well-defined metrics for the evaluation of specific quality characteristics. In this respect, García et al. [13] propose a framework for managing, modeling and evaluating software processes; define and experimentally validate a set of measures to assess, among others understandability and modifiability of process models. Similar empirical validation is provided by Sánchez-González et al. [25], who relate understandability and modifiability to innate characteristics of business process models.

Our approach differs from the above-mentioned ones in that we specifically focus on the process perspective of ETL processes. Instead of providing some characteristics as examples like in [28], we propose a comprehensive list of quality characteristics and we adjust them for our case. In addition, for each of these characteristics we provide quantitative metrics that are backed by literature.

	Characteristic	Barbacci et al. [5]	Simitsis et al. [28]	Jarke et al. [15]	Pavlov [24]	Naumann [23]	Dustdar et al. [10]	Kimball[2]
Characteristics with construct implications	**data quality**	-	data characteristics	quality dimensions	-	relevancy, reputation	data quality	data quality
	data accuracy			data accuracy		accuracy	accuracy	
	data completeness			data completeness		completeness	completeness	
	data freshness		freshness	data freshness, timeliness		timeliness	timeliness	
	data consistency		consistency	data coherence, correctness, minimality		consistent representation	consistency	consistency, deduplication, data conformance
	data interpretability			interpretability		interpretability		
	performance	performance	performance	performance, software efficiency	performance efficiency	-	performance	-
	time efficiency	latency	latency		time behaviour	latency, response time		paralleling & pipelining
	resource utilization				resource utilization			change data capture
	capacity	capacity, throughput				quality of service		
	modes	modes						types of fact tables
	cost efficiency	-	cost, affordability	-	-	value-added, price	pricing	-
	upstream overhead	-	overhead of source systems	-	upstream overhead	-	-	-
	security	security	-	security	-	security	security	security
	confidentiality	confidentiality						compliance management
	integrity	integrity						
	availability	availability	availability	availability			availability	availability
	auditability	-	auditability	-	auditability	-	provenance	lineage & dependency
	traceability		traceability	traceability	traceability	documentation		self-documenting
	reliability	reliability	reliability	reliability	reliability	-	reliability	reliability
	process availability	availability		responsiveness	availability	availability		
	fault tolerance	fault tolerance			fault tolerance			
	robustness	integrity	robustness		robustness			
	recoverability		recoverability		recoverability			recoverability, problem escalation
Characteristics for design evaluation	**adaptability**	-	-	-	adaptability	-	-	-
	scalability		scalability		scalability			scalability
	flexibility		flexibility					
	reusability			portability	reusability			
	usability	-	-	usability	-	-	-	visibility
	understandability					concise representation, understandability		understanding source data
	manageability	-	-	-	modularity, analyzability	-	-	manageability
	maintainability	maintainability	maintainability	maintainability	maintainability			maintainability
	testability			validation	testability	verifiability		
n/a		safety		accessibility, usefulness, believability		customer support, believability, objectivity, amount of data	licencing	

Fig. 2. ETL Process Characteristics

3 Extracting Quality Characteristics

Our model mainly derives from a systematic literature review that we conducted, following the guidelines reported by Kitchenham et al. [18]. The research questions addressed by this study are the following:
RQ1) What ETL process quality characteristics have been addressed?
RQ2) What is the definition for each quality characteristic?

Our search process used an automated keyword search of SpringerLink [2], ACM Digital Library [3], ScienceDirect [4] and IEEE Xplore [5]. The search strings were the following:

– (quality attributes OR quality characteristics OR qox) AND ("etl" OR "extraction transformation loading") AND ("information technology" OR "business intelligence")
– (quality attributes OR quality characteristics OR qox) AND ("data integration" OR "information systems integration" OR "data warehouses") AND ("quality aware" OR "quality driven")

The inclusion criterion for the studies was that they should identify a wide range of quality characteristics for data integration processes and thus only studies that mentioned at least 10 different quality characteristics were included. The quality characteristics could refer to any stage of the process as well as to the quality of the target repositories as a result of the process. One exclusion criterion was that studies should be written in English. Moreover, whenever multiple studies from same researcher(s) and line of work were identified, our approach was to include only the most relevant or the most recent study.

The result of our selection process was a final set of 5 studies. Nevertheless, in an attempt to improve the completeness of our sources, we also considered the ETL subsystems as defined in [2] for an industry perspective on the area, as well as standards from the field of software quality. Regarding software quality, our approach was to study the work by Barbacci et al. [5] and include in our model all the attributes relevant to ETL processes, with the required definition adjustments. This way we reviewed a commonly accepted, generic taxonomy of software quality attributes, while at the same time avoiding the adherence to more recent, strictly defined standards for practical industrial use, which we are nevertheless aware of [4]. The complete list from the resulting 7 sources, covering the most important characteristics from a process perspective that are included in our model can be seen in Fig. 2.

Data quality is a prime quality chracteristic of ETL processes. Its significance is recognized by all the approaches presented in our selected sources, except for Pavlov [24] and Barbacci et al. [5]. since the factors in their analyses derive directly or indirectly from generic software quality attributes. Our model was

[2] `http://link.springer.com`
[3] `http://dl.acm.org`
[4] `http://www.sciencedirect.com/`
[5] `http://ieeexplore.ieee.org`

enriched with a more clear perspective of data quality in Information Systems and a practical view of how quality criteria can lead to design decisions, after reviewing the work by Naumann [23].

Performance, was given attention by all presented approaches , which was expected since time behaviour and resource efficiency are the main aspects that have traditionally been examined as optimization objectives. On the other hand, the works of Pavlov [24] and Simitsis et al. [28] were the only approaches to include the important characteristic of *upstream overhead*. However, [24] does not include *security*, which is discussed in the rest of the works. The same is true for *auditability*, which is absent from the work of Barbacci et al. [5] but found in all other works. Reliability on the other hand, is recognized as a crucial quality factor by all approaches. As expected, the more abstract quality characteristics *adaptability* and *usability* are less commonly found in the sources, in contrast with *manageability* which is found in all approaches except for Dustdar et al. [10], who do not discuss about intangible characteristics.

Although we include *cost efficiency* in Fig. 2, in the remainder of this paper this characteristic is not examined as the rest. The reason is that we view our quality-based analysis in a similar perspective as Kazman et al. [17] , according to which any quality attribute can be improved by spending more resources and it is a matter of weighting the benefits of this improvement to the required cost that can lead to rational decisions. In addition, we regarded *safety* as non-relevant for the case of ETL processes, since these processes are computer-executable, non-critical and hence the occurrence of accidents or mishaps is not a concern. Similarly, we considered that the characteristics of *accessibility*, *usefulness*, *customer support*, *believability*, *amount of data* and *objectivity* found in [15] and [23] are not relevant for our case, as they refer to the quality of source or target repositories, yet do not depend on the ETL process. Likewise, *licencing* [10] refers to concrete tools and platforms while our ETL quality analysis is platform independent.

Through our study we identified that there are two different types of characteristics — characteristics that can actively drive the generation of patterns in the ETL process design and characteristics that cannot explicitly indicate the use of specific design patterns, but can still be measured and affect the evaluation of and the selection among alternative designs. In the remainder of this paper we refer to the first category as *characteristics with construct implications* and to the second as *characteristics for design evaluation.*

4 Process Characteristics with Construct Implications

In this section, we present our model for characteristics with construct implications. The proposed list of characteristics and measures can be extended or narrowed down to match the requirements for specific use cases.

4.1 Characteristics and Measures

In this subsection, we provide a definition for each characteristic as well as candidate metrics under each definition, based on existing approaches that we discovered coming from literature and practice in the areas of Data Warehousing and Software Engineering. For each metric there is a definition and a symbol, either (+) or (−) denoting whether the maximization or minimization of the metric is desirable, respectively.

1. *data quality*: the fitness for use of the data produced as the outcome of the ETL process. It includes:
   (a) *data accuracy*: percentage of data without data errors.
      M1: % of correct values [6] (+)
      M2: % of delivered accurate tuples [6] (+)
   (b) *data completeness*: degree of absence of null values and missing values.
      M1: % of tuples that should be present at their appropriate storage but they are not [27, 6] (−)
      M2: % of non-null values [6] (+)
   (c) *data freshness*: indicator of how recent data is with respect to time elapsed since last update of the target repository from the data source.
      M1: Instant when data are stored in the system - Instant when data are updated in the real world [6] (−)
      M2: Request time - Time of last update [6] (−)
      M3: 1 / (1 - age * Frequency of updates) [6] (−)
   (d) *data consistency*: degree to which each user sees a consistent view of the data and data integrity is maintained throughout transactions and across data sources.
      M1: % of tuples that violate business rules [27, 6] (−)
      M2: % of duplicates [6] (−)
   (e) *data interpretability*: degree to which users can understand data that they get.
      M1: # of tuples with interpretable data (documentation for important values) [6] (+)
      M2: Score from User Survey (Questionnaire) [6] (+)
2. *performance*: the performance of the ETL process as it is implemented on a system, relative to the amount of resources utilized and the timeliness of the service delivered. It includes:
   (a) *time efficiency*: the degree of low response times, low processing times and high throughput rates.
      M1: Process cycle time [21] (−)
      M2: Average latency per tuple in regular execution [27] (−)
      M3: Min/Max/Average number of blocking operations [27] (−)
   (b) *resource utilization*: the amounts and types of resources used by the ETL process.
      M1: CPU load, in percentage of utilization [21] (−)
      M2: Memory load, in percentage of utilization [21] (−)

(c) *capacity*: the demand that can be placed on the system while continuing to meet time and throughput requirements.
  M1: Throughput of regular workflow execution [27] (+)

(d) *modes*: the support for different modes of the ETL process based on demand and changing requirements, for example batch processing, real-time event-based processing, etc.
  M1: Number of supported modes / Number of all possible modes (+)

3. *upstream overhead*: the degree of additional load that the process causes to the data sources on top of their normal operations.
  M1: Min/Max/Average timeline of memory consumed by the ETL process at the source system [27] (−)
4. *security*: the protection of information during data processes and transactions. It includes:
  (a) *confidentiality*: the degree to which data and processes are protected from unauthorized disclosure.
    M1: % of mobile computers and devices that perform all cryptographic operations using FIPS 140-2 cryptographic modules [9] (+)
    M2: % of systems (workstations, laptops, servers) with latest antispyware signatures [1] (+)
    M3: % of remote access points used to gain unauthorized access [9] (−)
    M4: % of users with access to shared accounts [9] (−)
  (b) *integrity*: the degree to which data and processes are protected from unauthorized modification.
    M1: % of systems (workstations, laptops, servers) with latest antivirus signatures [1] (+)
  (c) *reliability*: the degree to which the ETL process can maintain a specified level of performance for a specified period of time. It includes:
    i. *availability*: the degree to which information, communication channels, the system and its security mechanisms are available when needed and functioning correctly.
      M1: Mean time between failures (MTBF) [27] (+)
      M2: Uptime of ETL process [27] (+)
    ii. *fault tolerance*: the degree to which the process operates as intended despite the presence of faults.
      M1: Score representing asynchronous resumption support [27] (+)
    iii. *robustness*: the degree to which the process operates as intended despite unpredictable or malicious input.
      M1: Number of replicated processes [27] (+)
    iv. *recoverability*: the degree to which the process can recover the data directly affected in case of interruption or failure.
      M1: Number of recovery points used [27] (+)
      M2: % of successfully resumed workflow executions [27] (+)
      M3: Mean time to repair (MTTR) [27] (−)
  (d) *auditability*: the ability of the ETL process to provide data and business rule transparency. It includes:

i. *traceability*: the ability to trace the history of the ETL process execution steps and the quality of documented information about runtime.
M1: % of KPIs that can be followed, discovered or ascertained by end users [20] (+)

Referring to our running example from Fig. 1, the difference between the first and the second design is that the latter includes an additional task for loading state's abbreviations (e.g., CA for California). This way less records would remain without state only because abbreviation would not be recognized. The second design additionally includes one task for checking the referential integrity constraint between the two dimension tables of the data warehouse. Consequently, the *Data Quality* for the second design is improved compared to the first one, in terms of *data completeness* and *data consistency*. This could be demonstrated by the measures *% of non-null values* and *% of tuples that violate business rules*, respectively.

4.2 Characteristics Relationships

In the same direction as [28] and [5] we also recognise that ETL process characteristics are not independent of each other and each time a decision has to be made, the alternative options might affect different characteristics differently, but that this is not realized in completely ad hoc ways. On the contrary, we argue that there is an inherent relationship between characteristics and it can be depicted in a qualitative model that can be instantiated per case and facilitate reasoning and automation.

Our model for the dependencies among characteristics with construct implications can be seen in Fig. 3 . In this model we include all the characteristics with construct implications that we have identified and defined in Sec. 3.

Our model consists of first-level characteristics and in some cases second- or even third-level sub-characteristics and can be read in a cause-and-effect fashion, i.e., improving one characteristic leads to improvement or deterioration of another characteristic. We should notice that although traditionally availability

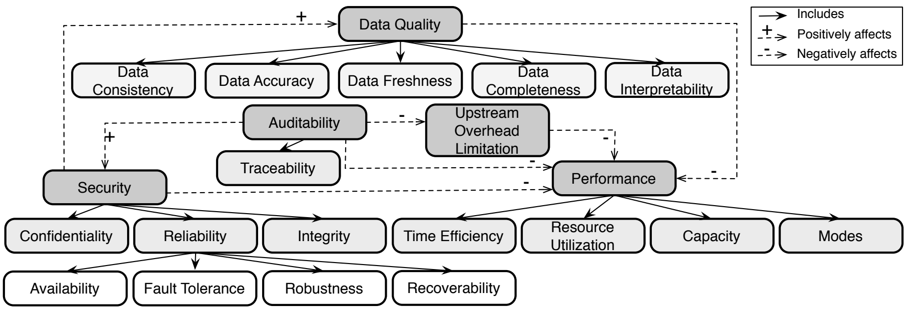

Fig. 3. Dependencies among process characteristics with construct implications

is classified directly under security, for our case availability is in fact a subgoal of reliability. In other words, reliability requires not only the satisfaction of availability but also maintaining specified SLAs for the ETL process and that justifies our decision to place availability under reliability and reliability under security.

Coming back to our running example from Fig. 1, it is clear that the second design would require more time and more computational resources than the first one in order to perform the additional tasks. The measures of *Process execution time* and *CPU load measured in percentage of utilization* would have higher values indicating worse *time efficiency* and *resource utilization*. Thus, improved *Data Quality* would have to be considered at the price of decreased *Performance* and whether or not the decision to select the second design would be optimal, would depend on the importance of each of these characteristics for the end-user.

As can be seen in Fig. 3 the improvement of any other characteristic negatively affects performance. That is reasonable since such improvements would require the addition of extra complexity to the ETL process, diverging from the optimal simplicity that favours performance. Improving Data Quality would require additional checks, more frequent refreshments, additional data processing and so on, thus utilizing more resources and imposing a heavier load on the system. In the same manner, improving security would require more complex authentication, authorization and accounting (AAA) mechanisms, encryption, additional recovery points, etc., similarly having negative impact on performance. Likewise, improving auditability would require additional processes for logging, monitoring as well as more resources to constantly provide real-time access to such information to end-users. In a similar fashion, promoting upstream overhead limitation would demand locks and scheduling to minimize impact of ETL processes on competing resources and therefore time and throughput limitations.

On the other hand, improving security positively affects data quality since data becomes more protected against ignorant users and attackers, making it more difficult for data and system processes to be altered, destroyed or corrupted. Therefore, data integrity becomes easier to maintain. In addition, improved system availability and robustness leads to improved data quality in the sense that processes for data refreshing, data cleaning and so on remain undisrupted.

Regarding the impact that improving auditability has to security, it is obvious that keeping track of system's operation traces and producing real-time monitoring analytics foster faster and easier threat detection and mitigation, thus significantly benefiting security. On the contrary, these operations have a negative impact on upstream overhead limitation, following the principle that one system cannot be measured without at the same time being affected.

5 Process Characteristics for Design Evaluation

In this section we show our model for characteristics for design evaluation. As in Sec. 4 we first define the characteristics and then show the relationships among them.

5.1 Characteristics and Measures

Following the same approach as with characteristics with construct implications, in this subsection, we provide a definition for each characteristic for design evaluation, as well as proposed metrics deriving from literature.

1. *adaptability*: the degree to which ETL process can effectively and efficiently be adapted for different operational or usage environments. It includes:
   (a) *scalability*: the ability of the ETL process to handle a growing demand, regarding both the size and complexity of input data and the number of concurrent process users.
      M1: Ratio of system's productivity figures at two different scale factors, where productivity figure = throughput * QoS/ cost [16] (+)
      M2: # of Work Products of the process model [13] (−)
   (b) *flexibility*: the ability of the ETL flow to provide alternative options and dynamically adjust to environmental changes (e.g., by automatically switching endpoints).
      M1: # of precedence dependences between activities [13] (−)
   (c) *reusability*: the degree to which components of the ETL process can be used for operations of other processes.
      M1: # of dependences between activities with locality (e.g., in the same package) [12] (+)
      M2: # of dependences between activities without locality (e.g., from different packages) [12] (−)
      The following measures are valid in the case where there are statistical data about the various modules (e.g., transformation or mapping operations) of the ETL process:
      M3: % of reused low level operations in the ETL process [12] (+)
      M4: Average of how many times low level operations in the ETL process have been reused per specified time frame [12] (+)
2. *usability*: the ease of use and configuration of the implemented ETL process on the system. It includes:
   (a) *understandability*: the clearness and self-descriptiveness of the ETL process model for (non-technical) end users.
      M1: # of activities of the software process model [13] (−)
      M2: # of precedence dependences between activities [13] (−)
3. *manageability*: the easiness of monitoring, analyzing, testing and tuning the implemented ETL process.
   (a) *maintainability*: the degree of effectiveness and efficiency with which the ETL process can be modified to implement any future changes.
      M1: Length of process workflow's longest path [27] (−)
      M2: # of relationships among workflow's components [27] (−)
      M3: Cohesion of process workflow (viewed as a directed graph) [8] (+)
      M4: Coupling of process workflow (viewed as a directed graph) [8] (−)
      M5: # of input and output flows in the process model [22] (−)
      M6: # of output elements in the process model [22] (−)

M7: # of merge elements in the process model [22] (−)
M8: # of input and output elements in the process model [22] (−)

(b) *testability*: the degree to which the process can be tested for feasibility, functional correctness and performance prediction.
M1: Cyclomatic Complexity of the ETL process workflow [14] (−)

Regarding our running example from Fig. 1, we can clearly see how the second design is less usable since it is less understandable. This is not only an intuitive impression from looking at a more complex process model but can also be measured using the measures of *# of activities of the process model*, which is greater for the second design.

5.2 Characteristics Relationships

In Fig. 4 we show the dependencies among characteristics for design evaluation. Increased usability favours manageability because a more concise, self-descriptive system is easier to operate and maintain. Similarly, adaptability positively affects usability, since an easily configured system is easier to use and does not require specialized skill-set from the end user. On the other hand, adaptability can be achieved with more complex systems and therefore it negatively affects manageability. This negative relationship might appear counter-intuitive, but it should be noted that our view of adaptability does not refer to autonomic behaviour, which would possibly provide self-management capabilities. Instead, we regard manageability from an operator's perspective where control is desirable and the addition of unpredictable, "hidden" mechanisms would make the process more difficult to test and maintain. Regarding the apparent conflict between the negative direct relationship among Adaptability and Manageability and the transitive positive affection of Adaptability–Usability–Manageability, this can be explained by the different effect of each influence, which can be considered as differing weights on the edges of the Digraph.

Going back to our running example from Fig. 1, it is apparent that the first design is easier to manage, since it is easier to maintain. This can be verified using any of the metrics for maintainability defined in this section. Thus, we can see how *usability* positively impacts *manageability*.

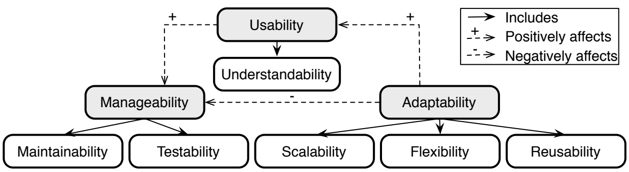

Fig. 4. Dependencies among characteristics for design evaluation

6 Summary and Outlook

The automation of ETL processes seems as a promising direction in order to effectively face emerging challenges in Business Intelligence . Although information systems are developed by professionals with technical expertise, it is important to orientate the design of underlying processes in an end-user perspective that reflects business requirements. In this paper, we have proposed a model for ETL process quality characteristics that constructively absorbs concepts from the fields of Data Warehousing, ETL, Data Integration and Software Engineering. One important aspect about our model is that for each and every characteristic, there has been suggested measurable indicators that derive solely from existing literature. Our model includes in a high level the relationships between different characteristics and can indicate how improvement of one characteristic by the application of design modifications can affect others.

Our vision is that our defined models can be used as a palette for techniques that will automate the task of selecting among alternative designs. Future work will target the development of a framework that will use this model as a stepping stone to provide automatic pattern generation and evaluation for ETL processes, keeping quality criteria at the center of our analysis.

Acknowledgements. This research has been funded by the European Commission through the Erasmus Mundus Joint Doctorate "Information Technologies for Business Intelligence - Doctoral College" (IT4BI-DC). This work has also been partly supported by the Spanish Ministerio de Ciencia e Innovación under project TIN2011-24747.

References

[1] KPI library, `http://kpilibrary.com` (cited January 2014)
[2] The subsystems of ETL revisited, `http://www.informationweek.com/software/information-management/kimball-university-the-subsystems-of-etl-revisited/d/d-id/1060550` (cited January 2014)
[3] El Akkaoui, Z., Mazón, J.-N., Vaisman, A., Zimányi, E.: BPMN-based conceptual modeling of ETL processes. In: Cuzzocrea, A., Dayal, U. (eds.) DaWaK 2012. LNCS, vol. 7448, pp. 1–14. Springer, Heidelberg (2012)
[4] Al-Qutaish, R.: An investigation of the weaknesses of the ISO 9126 Intl. Standard. In: ICCEE, pp. 275–279 (2009)
[5] Barbacci, M., Klein, M., Longstaff, T., Weinstock, C.: Quality Attributes. Tech. rep., Carnegie Mellon University, Pittsburgh, Pennsylvania (1995)
[6] Batini, C., Cappiello, C., Francalanci, C., Maurino, A.: Methodologies for data quality assessment and improvement. ACM Comput. Surv. 41(3), 1–52 (2009)
[7] Böhm, M., Wloka, U., Habich, D., Lehner, W.: GCIP: Exploiting the generation and optimization of integration processes. In: EDBT, pp. 1128–1131. ACM (2009)
[8] Briand, L., Morasca, S., Basili, V.: Property-based software engineering measurement. IEEE Trans. on Soft. Eng. 22(1), 68–86 (1996)

[9] Chew, E., Swanson, M., Stine, K.M., Bartol, N., Brown, A., Robinson, W.: Performance Measurement Guide for Information Security. Tech. rep. (2008)

[10] Dustdar, S., Pichler, R., Savenkov, V., Truong, H.L.: Quality-aware service-oriented data integration: Requirements, state of the art and open challenges. SIGMOD 41(1), 11–19 (2012)

[11] El Akkaoui, Z., Zimanyi, E.: Defining ETL worfklows using BPMN and BPEL. In: DOLAP, pp. 41–48. ACM (2009)

[12] Frakes, W., Terry, C.: Software reuse: Metrics and models. ACM Comput. Surv. 28(2), 415–435 (1996)

[13] García, F., Piattini, M., Ruiz, F., Canfora, G., Visaggio, C.A.: FMESP: Framework for the modeling and evaluation of software processes. In: QUTE-SWAP, pp. 5–13. ACM (2004)

[14] Gill, G., Kemerer, C.: Cyclomatic complexity density and software maintenance productivity. IEEE Trans. on Soft. Eng. 17(12), 1284–1288 (1991)

[15] Jarke, M., Lenzerini, M., Vassiliou, Y., Vassiliadis, P.: Fundamentals of Data Warehouses. Springer (2003)

[16] Jogalekar, P., Woodside, M.: Evaluating the scalability of distributed systems. IEEE Trans. on Parallel and Distributed Systems 11(6), 589–603 (2000)

[17] Kazman, R., Asundi, J., Klein, M.: Quantifying the costs and benefits of architectural decisions. In: ICSE, pp. 297–306 (2001)

[18] Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., Linkman, S.: Systematic literature reviews in software engineering - a systematic literature review. Inf. Softw. Technol. 51(1), 7–15 (2009)

[19] van Lamsweerde, A.: Goal-oriented requirements engineering: a guided tour. In: Requirements Engineering, pp. 249–262 (2001)

[20] Leite, J., Cappelli, C.: Software transparency. Business and Information Systems Engineering 2(3), 127–139 (2010)

[21] Majchrzak, T.A., Jansen, T., Kuchen, H.: Efficiency evaluation of open source ETL tools. In: SAC, pp. 287–294. ACM, New York (2011)

[22] Muñoz, L., Mazón, J.N., Trujillo, J.: Measures for ETL processes models in data warehouses. In: MoSE+DQS, pp. 33–36. ACM (2009)

[23] Naumann, F.: Quality-Driven Query Answering for Integrated Information Systems. LNCS, vol. 2261. Springer, Heidelberg (2002)

[24] Pavlov, I.: A QoX model for ETL subsystems: Theoretical and industry perspectives. In: CompSysTech, pp. 15–21. ACM (2013)

[25] Sánchez-González, L., García, F., Ruiz, F., Mendling, J.: Quality indicators for business process models from a gateway complexity perspective. Inf. Softw. Technol. 54(11), 1159–1174 (2012)

[26] Simitsis, A., Vassiliadis, P., Sellis, T.: Optimizing ETL processes in data warehouses. In: ICDE, pp. 564–575 (2005)

[27] Simitsis, A., Vassiliadis, P., Dayal, U., Karagiannis, A., Tziovara, V.: Benchmarking ETL Workflows. In: Nambiar, R., Poess, M. (eds.) TPCTC 2009. LNCS, vol. 5895, pp. 199–220. Springer, Heidelberg (2009)

[28] Simitsis, A., Wilkinson, K., Castellanos, M., Dayal, U.: QoX-driven ETL design: Reducing the cost of ETL consulting engagements. In: SIGMOD, pp. 953–960. ACM, New York (2009b)

[29] Wilkinson, K., Simitsis, A., Castellanos, M., Dayal, U.: Leveraging business process models for ETL design. In: Parsons, J., Saeki, M., Shoval, P., Woo, C., Wand, Y. (eds.) ER 2010. LNCS, vol. 6412, pp. 15–30. Springer, Heidelberg (2010)