Data Quality Problems in ETL: The State of the Practice in Large Organisations

PHILIP WOODALL, University of Cambridge
ALEXANDER BOREK, Gartner
MARTIN OBERHOFER, IBM
JING GAO, University of South Australia

This paper presents a review of the data quality problems that arise because of Extract, Transform and Load (ETL) technology in large organisations by observing the context in which the ETL is deployed. Using a case study methodology, information about the data quality problems and their context arising from deployments in six large organisations is reported. The findings indicate that ETL deployments most commonly introduce data accessibility problems which are caused by (1) the ETL failing part way and not delivering the data on time, (2) the information systems being locked during ETL execution, and (3) users not being able to find data in the target because of errors in the way the primary keys are transformed. Furthermore, accuracy, timeliness, believability, and representational consistency problems were also found to be caused by the ETL technology.

1. INTRODUCTION

With the ever-growing digitalization of our economy, the dependence of business operations and decision making on deriving new insights from data is constantly increasing [Davenport and Harris 2007]. An essential requirement for this is that data needs to be with decision makers, analysts and managers when and where they need it [Smith et al. 2008], and one of the well-established enabling technologies for transferring data is Extract, Transform and Load (ETL). However, ETL is not a simple technology to implement, and while it can be used to address data quality problems [Kimball and Caserta 2004], it can also yield data quality problems in different business settings.

This paper surveys the ways in which ETL is actually used in organisations to determine the data quality problems that arise from this use. The results can be used to determine what data quality problems future ETL solutions (and other data integration technologies) must address to be most useful to organisations. Other survey research into ETL has focussed on the technology itself and how it can be improved with respect to the state of the art [Vassiliadis 2009] or has focussed on the ETL-related tools [Thomsen and Pedersen 2005; Thoo and Randall 2015]. Our focus is different: this paper takes a state of the practice viewpoint and our particular focus is on identifying the data quality problems that are caused because of the way the ETL is used, implemented and configured (i.e. because of the particular context in which it is deployed) in an organisation. It is important to note that we do not intend to report the data quality problems that may already exist in the source databases and that may be simply transferred by the ETL to the target; it is the problems that are introduced by the ETL itself which we aim to identify.

ETL use and the context of the business is likely to change dramatically with different sizes of business (e.g. small vs very large organisations), and this research therefore chooses to focus on only large organisations. In order to determine what data quality problems are caused by ETL in large organisations, this paper first answers the sub questions:

1. What is ETL used for, and why is it used like this in the organisation?
2. How and why is it configured/used like this?

Using a case study approach with findings from six different organisations, the results indicate that data accessibility problems are most common followed by accuracy, timeliness, believability and representational consistency problems.

2. EXISTING ETL RESEARCH

Existing work transcending ETL and data quality focusses on how the ETL technology can be used to address data quality issues (see for example, [Rodic and Baranovic 2009] [Galhardas et al. 2001]). Furthermore, existing research describes some of the problems caused by the ETL itself: these include data unavailability/lack of freshness caused by the ETL not finishing on time because of a long execution time or a failure in the ETL which causes it to need to be run again [Simitsis et al. 2005; Simitsis et al. 2010]. In the future, new uses of ETL, such as in data lakes (see Figure 1) are likely to put further demands on ETL technologies as they become part of the core business—especially with regards to automation: much of the ETL is still manual, and a current limitation of the self-service approach is that the commercial tools have limited support for automatically generating and including complex data cleansing processes into the data movement logic [Maier et al. 2012].

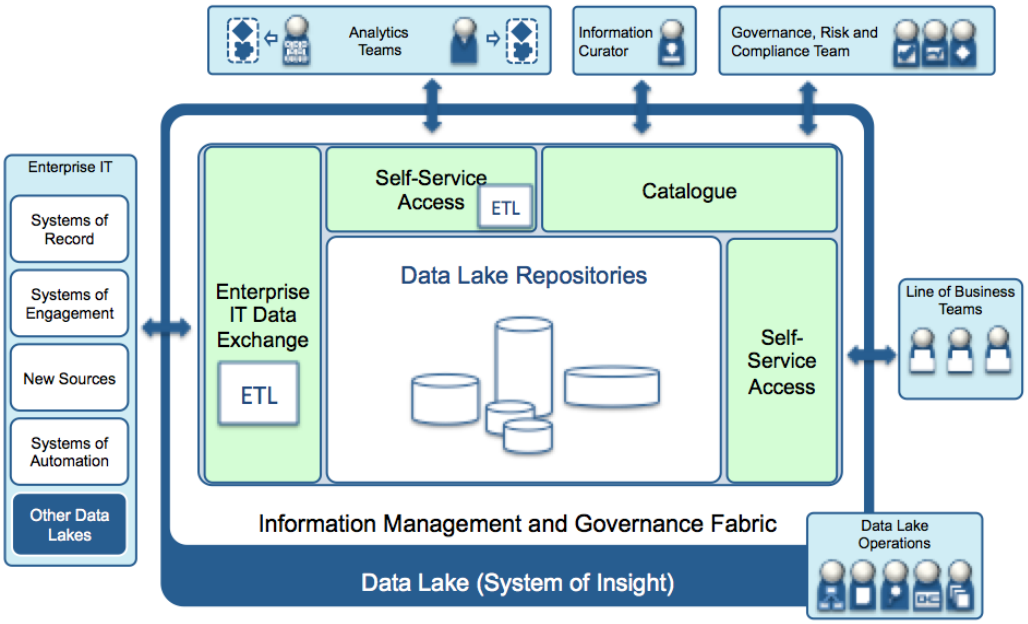

Figure 1: Future uses of ETL in a data lake including self-service access

Gartner lists five data integration scenarios [Thoo and Randall 2015]: Data acquisition for business intelligence (BI), analytics and data warehousing (A); sourcing and delivery of master data in support of master data management (MDM) (B); data migrations/conversions (C); Data consistency between operational applications (D); and Interenterprise data sharing (E). The analysis of the results from the case studies uses these to classify the ETL scenarios found.

3. METHODOLOGY

In order to answer the research question, a case study approach was applied which utilised, primarily, interviews and direct observations [Yin 2009]. A detailed inspection

of the various different cases was obtained to fully understand the context and determine the actual data quality problems present because of the context. For the selection of organisations to inspect, a convenience sample was used based on the different companies that the authors had access to (either by conducting a consulting or research project at the company, or by having access to experts from the organisation who were available to answer the questions). The only selection criteria applied was that the organisations were required to have more than $1B revenue. In the case of the healthcare organisation, it is a national service for a particular country and so is large in scale. The data collection was designed to be completely confidential (no names of experts or companies are revealed) because this gave the best chances of obtaining uncensored answers; especially as respondents were asked to report problems/failures. The ways in which the data extraction was carried out for each case are summarised in Table I.

Table I. Selected cases and the primary data extraction method used

Case	Business Sector	Primary Data Extraction Method
1	Manufacturing	Multiple telephone interviews with a user of the related IT systems
2	Manufacturing	Multiple interviews with various users of the IT systems
3	Retail Bank	Direct observation via a consulting project
4	Financial Asset Manager	Direct observation via a consulting project
5	Healthcare Authority	Multiple interviews
6	Wholesale Distributor	Direct observation via a consulting project and telephone interviews with a user of the IT systems

4. RESULTS: STATE OF THE PRACTICE ETL USE, CONFIGURATION AND DATA QUALITY PROBLEMS

The following subsections describe in-detail the results for each organisation (case). The descriptions are structured in terms of what the ETL is used for and why, how and why it is configured the way it is, and finally the data quality problems associated with the ETL processes. Using the conceptual framework of data quality (see figure 2 in [Wang and Strong 1996]), the data quality problems are classified in accordance with the dimensions in that framework.

4.1 Case 1: Manufacturer

What is it used for, and why is it used like this?

The manufacturer exports data from their ERP system (SAP in this case) into bespoke reporting tools via ETL because the ERP system does not provide the reporting fidelity required. In particular, reports are used, to determine what engineering parts have increased in volume over the last year, what parts are late (have not yet been delivered when expected), and to predict what parts are going to be late, etc. Furthermore, it is necessary to observe these reports on a detailed part-by-part basis as well as an aggregated view. The main problem is that the ERP system cannot provide a convenient way to view and manipulate aggregations of parts. One of the reporting tools used is IBM's Cognos, which helps to satisfy the aggregation reporting requirements. The data volume is large as the entire bill of materials (BOM) for the various aerospace products that the company manufactures is contained in the ERP system. Some sub-components have in excess of 300 parts, and so overall total for the final product and for all of the different types of products is large. The exact number of parts cannot be revealed because of confidentiality reasons.

How and why is it configured/used like this?
The ETL executes every Sunday and takes at least 12 hours to complete the transfer of all the data. Batch loading is used where all the data from the ERP system is extracted and loaded into the target reporting systems. The reporting systems are used from Monday to Friday by various users, as is the ERP system. The developers of the ETL chose to use the most reliable and uncomplicated method of transferring the data to ensure that the data is always available (i.e. to minimise the likelihood of data transfer failure). It is known that their method of batch loading sacrifices data timeliness, but this is a trade-off that is accepted. Other options like incremental loading are not used because of their complexity and therefore increased likelihood of failure. The complexity referred to is the need to develop code to determine what data needs updating, which is relatively easy compared to the additional need to ensure the integrity of the transactions in the target systems (for example, to ensure no duplicate records or incorrect deletion of records which should still be present).

What are the problems?
Clearly the main problem is the timeliness of the data with the reporting tools only being up to date on Monday morning, and with the data becoming increasingly stale throughout the week. Note that the ERP system is always kept up to date throughout the week, and, importantly, there are many updates to the system which should ideally be transferred to the reporting tools as soon as possible. Another problem that occurred once, according to the respondent, was that the ETL failed and the data did not transfer correctly. The result was that the reporting system was unavailable for a whole day while the data transfer was re-run. This is classified as an accessibility problem.

4.2 Case 2: Manufacturer

What is it used for, and why is it used like this?
In this case, the manufacturer transfers data via ETL between various operational systems in order to complete the business process of procurement of its engineering parts. For instance, an engineer will request a particular part to be procured and this enters a queue in the procurement agents' system. Once the procurement agent has obtained quotations from suppliers, the quotations are transferred via ETL to the purchase ordering system so that the order can be placed. There are also other ETL processes that transfer data from the purchase order system to various other systems, such as a final order release system, which can send an order to a supplier. Various COTS systems were procured and installed because the company has not found one system that can deal with all of the scenarios in procurement (e.g. buying ordinary consumable items individually from a retailer and buying hundreds of thousands of specialist parts via a contract with a specific supplier over a long period). Hence, a different system is needed for each scenario that existing systems cannot cater for. They currently have at least 5 different systems which enable procurement agents to buy parts for various scenarios.

How and why is it configured/used like this?
All the ETL processes in this case are batch processes. Note that some of these may act as an incremental load if the source system maintains a queue of orders that are removed from the system as they are transferred to another system. Most ETL processes are executed "out of working hours" while some are executed in the afternoon.

Some legacy systems don't support real-time updating, and the system is locked while the data is extracted/loaded. Therefore, in these cases, ETL "out of working hours" is preferable as it does not lock users out of the systems when they need access to them. Furthermore, batch loading is a simple and fast way to implement data transfer between these systems, and so that is what is used rather than incremental loading, which is more complex

What are the problems?
If the ETL process is executed during working hours, then it often effectively "locks" the systems as it extracts and loads data, resulting in the systems and data within being unavailable for decision makers until they are released. We observed this situation in one organisation where staff must stop their work on Friday afternoons while a batch of data is sent via ETL from one system to another. The reason was because of computing power which was being taken up with the ETL load process, and so other systems would slow down to a point of becoming unusable (and effectively "locked"). This is an accessibility data problem (rather than a timeliness problem) because the data exists in the system in an up-to-date fashion but it cannot be accessed. Furthermore, the different source systems reference the same data product (e.g. a work order) in different ways (for example, in one system a "project code" is used and in another system "buyer id" is used for work orders). When data is transferred to another system, it becomes a problem to find work orders for people who use a different identifier to the one selected by the ETL in the target system (hence this is also an accessibility problem). There are also various problems related to translation errors, synchronization errors and issues about the timing of updates, which result in inconsistencies across systems. Hence it is sometimes difficult for the procurement agents and planners to be sure if the data is correct or not. The relevant data quality dimension for this latter point is believability (due to the staff not be confident in the data).

4.3 Case 3: Retail Bank

What is it used for, and why is it used like this?
The retail bank uses ETL for integrating data from over 50 source systems to create a 360 degree view of the customer. This includes customer master data (such as address, birth date, phone number etc.), all contracts and products that the customer has with the bank, a history of interactions across all possible channels (e.g. physical branch, mobile app banking, online website banking, call center etc.), risk scorings, advertisements and promotions sent out to this customer, and many other pieces of relevant data. IBM DataStage is used as the ETL tool alongside custom made SQL scripts. The data is transferred via batch runs from the source systems during the night. The integrated data is used by many stakeholders in the bank. For example, it is used to create performance reports for marketing and sales using a combination of off the shelf reporting tools, namely IBM Cognos and SAS. The 360 customer view data is also used as input for self-service Business Intelligence (BI) and discovery tools, where additional ETL processes are run on a desktop environment of the business users to integrate additional data sources on an ad-hoc basis and to make further data transformations. Tools used by the business users for this purpose are, for instance, Qlikview, TIBCO Spotfire, and SAS. Finally, this data is also used as input for data mining and advanced analytics model building for sales and marketing by data scientists working on the business side, who also run further ETL processes on their

machines. Such models are used, for example, to identify customers that are most likely to buy.

How and why is it configured/used like this?
Every night an incremental load executes and copies only the changes made during the day. A full copy of the data is made once a week on Sunday, as it takes at least eight hours for the full batch run to complete. The daily incremental load ensures that data is up to date every day, while the weekly full run is needed to fix data inconsistencies that are a result of the incremental load during the week. In the past, real-time data was not a business requirement, which is most likely to change in the near future.

What are the problems?
A problem that occurs frequently is that the ETL process is interrupted because the data loaded from the sources does not meet the specifications. The ETL process is in some cases very intolerant to any inconsistencies in the source data. The result is that the process needs to be restarted after the source data has been corrected, which causes long delays for the data users. This can be caused because of 1) data defects in the source systems, e.g. if the data has not been entered accurately by the personnel in the branch, and 2) the source systems are not able to provide the data required in time, which has an impact on the start time of the ETL processes (this happens less frequently). Ultimately, these cause a data accessibility problem as users of the target system need to wait for the data.

4.4 Case 4: Financial Asset Manager

What is it used for, and why is it used like this?
A data warehouse is used to calculate a weekly view of the market risk of all financial assets owned by the investment management company. Therefore, data about all trading activities during the day and the current financial assets owned (quantity and market value) is extracted from 12 different trading and operational systems. Then, the market risk is calculated by simulating different events and volatilities that could happen in the market and generating an evaluation for each asset class and sub-class. The market risk figures are the key input for risk management and to inform the customers of the asset manager about the risk evaluations of their financial assets. The data can be analysed with a custom written business intelligence tool after further transformation using self-service BI.

How and why is it configured/used like this?
There is a weekly batch run to transfer the data, which takes half a day and runs during the night. At the end of the ETL transfer, the simulations are run to calculate the market risk evaluations. The ETL process uses self-coded SQL scripts and moves the data into an ODBC database platform. The data is transformed into a common format and an asset class and a sub-class are assigned to each financial asset. The amount of input data is limited before the simulation run (less than 1 GB), but the simulated data has a higher volume (>5 GB). After completion of the ETL process, a number of plausibility checks are run before the data is released. There is currently no requirement and business need to calculate the market risk more often. Hence, there needs to be a significant benefit if the system should be changed to daily ETL process runs.

What are the problems?
In some cases, financial investments are mapped to a wrong asset class during the ETL, which results in an inaccurate risk evaluation. This is clearly an accuracy problem in the data which causes any calculations to be inaccurate also (i.e. "garbage in garbage out"). The problem can be detected if there is a large change in the daily market risk value that cannot be explained with events in the real world, this often indicates that some parts of the simulations and/or calculations have gone wrong.

4.5 Case 5: Healthcare Authority

What is it used for, and why is it used like this?
In this case, the healthcare organisation's health funding model is being changed to an activity-based funding model, and BI and reporting are key enablers. The state department has, therefore, increasing demand to harvest data from various public care services providers and local hospitals etc. on a daily basis. This data is used for reporting purposes, for example to report the performance of all the local health services. The performance evaluation unit at the state level controls a data warehouse and runs regular reports to monitor local activities and alert the local health services if unusual behaviour is detected (e.g. they are over budget, too many surgeries of the same kind within a short period, unusual long waiting period for elective surgeries etc.). The BI setup involves a two-level data warehouse infrastructure. On a daily basis, data is extracted from systems such as patient admission systems and emergency service systems (at the local health districts), and is then transferred to the data warehouse at the local health district level. Data is then transferred via ETL to the data warehouse at the state department level from the local health districts, a data quality profiling engine scans all loaded data first.

How and why is it configured/used like this?
An incremental load ETL process between the various local health district data source systems and the local health data warehouse is run daily. Also, weekly incremental loading is scheduled between the local health district data warehouse and the state data warehouse. Incremental loading is used because of a slow network infrastructure which charges by data usage. Furthermore, because patient data is sensitive, incremental loading minimises the data volume that is exposed for any data breaches/releases. The autonomous nature of health service providers enables them to choose and implement IT systems themselves. Over time, the state-level data warehouse therefore plays an important role in integrating data from these heterogeneous and disparate source systems.

What are the problems?
The maintenance of data quality rules causes problems. For example, the ETL matches a treatment to a health coding system and the general dental code 101 contains the sub codes d01 and d02 (e.g. check-up and simple filling). With this ETL rule, everything is validated and loaded into the target. However, if a new rule is enforced that changes d01 to no longer be part of code 101, then the new rule will reject all 101/d01 records in the transformations of the ETL. Furthermore, the 101/d01 records that already exist in the data warehouse will no longer be correct and need to be updated. This is therefore both an accessibility problem (because of the failure of the ETL to transfer the data to the target) and an accuracy problem.

4.6 Case 6: Wholesale distributor

What is it used for, and why is it used like this?

The respondent mentioned two main examples of how ETL is used in the organisation: for sharing of operational data between applications, and as a one-off transfer for system migrations.

Sub case 1 (sharing operational data between applications): In the first case, the data is transferred from an ERP system (SAP in this case) into various target systems. One example is into a Master Data Management (MDM) system (via a batch transfer), which includes data for sellers (50-60K records periodically, with an annual batch update of approx. 300K records), and billing and accounts data (1000 records per day, the largest number being 50K records). Data is extracted from the ERP system so that it can be entered into the MDM solution and data stewards can, for example, ensure the integrity of the data including reconciling complex cases of duplicate records. Also, any non-core business activities that need data from the ERP system have it extracted so that the core activities (such as placement of orders) have maximum performance from the ERP system and are not burdened by other processes.

Sub case 2 (one-off system migrations):

Due to an upgrade in the platform used to host the company website, a migration from the old systems to the new platform was needed. This involved transferring the data via ETL to the new platform. This opportunity was taken to clean the data during the transforms. Although each migration is a one-off case, the respondent indicated that the company is constantly doing these; other examples besides the website include on-boarding new applications and integrating external data (such as from data analytics providers) into the organisation; or sending data to external cloud/Software as a Service (SaaS) solutions for external analysis (in this case, Salesforce). These are always done in batch and involve large amounts of data: monthly, 1.5 - 3 million records and annually approx. 24 million records for the external provider case.

How and why is it configured/used like this?

Sub case 1 (sharing operational data between applications): Data is mainly transferred periodically as the business need is for data is in batch updates rather than continuous real-time data. All data is extracted into a staging area which is then used to identify and to load only the changed data (i.e. batch extract and incremental load). The previous snapshot of the data is compared to the new data in the staging area in order to determine what data has changed. ETL is generally done in the evenings/overnight, so that daily business processes are not disrupted.

Sub case 2 (one-off system migrations): Clearly, for one-off migrations of data, real-time synchronisation is not needed, hence batch loading is done. When integrating data from external providers it is also done in batch because the data is provided in batches (e.g. a monthly subscription gives a monthly update of data). It is still possible to do an incremental load if the new monthly data is compared to the old data before loading and only new/updated data is loaded. However, an entire load is done annually as a refresh.

What are the problems?

Sub case 1 (sharing operational data between applications): The ERP system has validation rules that are different to the target systems' rules, so when the data is transferred to the MDM system and subsequently to the target systems, the data may have passed the ERP rules, but then fails the target system rules (so it is considered to be inaccurate in the target). Note that the MDM system is configured to be flexible

to allow any data to be entered (with the aim of checking and cleaning the data later). An on-going problem that has a workaround implemented to keep it functioning is the dropping of the connection to the ERP system when transferring data. This causes the ETL to fail and hence data to be unavailable (an accessibility problem).

Sub case 2 (one-off system migrations): Stale data (timeliness) is a problem when integrating data from external data providers because updates are only done on a monthly basis. Also, the inconsistencies in character sets between the source and target systems (representational consistency) have caused problems: in one case, Spanish and French characters were dropped during the transfer and so the target system contained erroneous data.

5. DISCUSSION AND CONCLUSION

The second column in table II shows that all scenarios (A-E) reported by Gartner (see section 2) were found in the cases. In terms of the data quality problems found, the most common problem is accessibility of data (see the summary of problems in table II and Figure 2). Accessibility problems are caused in three ways: the ETL failing and hence not delivering the data on time to the target system(s), the information systems being locked during ETL execution, and users not being able to find data in the target. The ETL could fail because of transformation problems (which include mostly data validation rule failures) and dropped connections. Even with the simplest form of ETL (the batch load), the ETL can fail. Failures are exacerbated by the fact that most of the ETL processes are executed out of working hours and so failures are often not discovered early. The information systems are locked when the ETL executes to allow it access to read and write data. Hence, users must wait until the ETL releases the lock before they can use the data. Another cause of accessibility problems is users not being able to find data in the target system because of inconsistencies introduced into the primary keys (by the ETL transformations) which are used to find data.

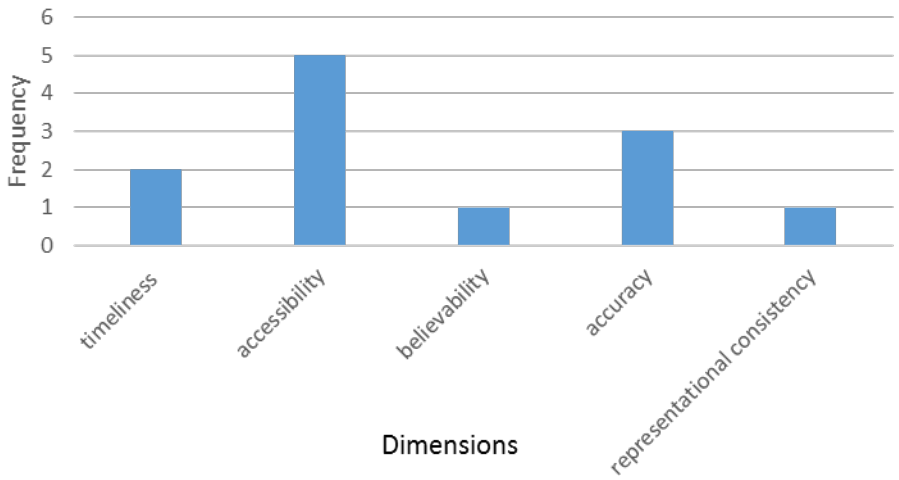

Figure 2: Frequencies of data quality problems occurring from ETL processes

Table II. Summary of the ETL case results

Case	What is it used for?	Why is it used?	How is it configured?	Why is it configured/ used like this?	What are the problems?
1	Reporting and analytics (A)	ERP does not have necessary reporting capabilities	Batch transfer, executed once per week	For simplicity and reliability, and once per week because (12+ hours to run)	Stale data as the week progresses (timeliness) Data not transferred (unknown ETL failure) (accessibility)
2	Connecting operational systems (D)	The company has not found one IT system that can deal with all of the scenarios in procurement	Batch transfer, executed overnight or once per week.	Staff are already unsure about the consequences of making changes, so want as simplistic ETL process as possible	Locked systems, and users cannot find data because of the ETL introducing inconsistent keys (accessibility) Staff are not confident in the data (believability)
3	Reporting and analytics (A) and MDM (B)	Customer data is found in multiple source systems	Incremental load (once per day and overnight). Batch load on Sunday	Incremental load keeps data up-to-date, and the batch load is needed to correct data inconsistencies that occur during the weekly loads	Data fails ETL validation rules and leaves users without data in the target (accessibility)
4	Transferring data to a data warehouse (A)	To integrate data and provide a target system on which to run simulations	Batch transfer, once per week overnight (takes approx. 12 hours)	No need to have the data more frequently	Incorrectly mapped values (accuracy)
5	Reporting and analytics (A)	To integrate data	Incremental load daily (local level) and incremental load once per week (state level)	Slow network infrastructure with charges linked to amount of data transferred. Minimises the data volume exposed for any data breaches/ releases	Changes in data quality rules render data in the target inaccurate (accuracy) and cause the ETL to fail (accessibility)
6.1	Connecting operational systems (D) and MDM (B)	To unburden the ERP system and transfer data into MDM software	Batch transfers (full extract and incremental load)	No need to have the data more frequently, and avoids inefficient data transfers	Data fails target system validation rules (accuracy). Dropped connections during ETL leaves data inaccessible (accessibility)
6.2	data migrations (C) and Inter-enterprise data sharing (E)	It is an simple way to transfer data in bulk	Batch transfers	Batch is the only possible option (external data is not available in real-time)	Infrequent data updates (timeliness) Inconsistencies in character sets between the source and target (representational consistency).

Inconsistencies also cause accuracy problems. These accuracy problems can be introduced by ETL processes either because the ETL transforms incorrectly map the data or because of differences in data validation rules between source and target systems. These can also lead to believability problems when users lose confidence in the data in the target system, or representational consistency problems with incorrectly mapped characters (for example, in different languages).

Timeliness problems are also caused by the ETL processes when they cannot be executed frequently enough to refresh the data as required. However, these are only problematic if the data is actually needed before the next data refresh. ETL solutions should therefore consider the refresh rate needed before developing solutions that attempt to refresh data more often than needed.

ACKNOWLEDGMENTS

The authors would like to thank all the respondents for their time and willingness to share their experiences.

REFERENCES

Y. Cui and J. Widom. 2003. Lineage tracing for general data warehouse transformations. *VLDB J.* 12, 1 (2003), 41–58. DOI:http://dx.doi.org/10.1007/s00778-002-0083-8

Thomas H. Davenport and Jeanne G. Harris. 2007. *Competing on Analytics: The New Science of Winning*, Harvard Business School Press.

Helena Galhardas, Daniela Florescu, Dennis Shasha, Eric Simon, and Cristian-Augustin Saita. 2001. Declarative Data Cleaning: Language, Model, and Algorithms. In *Proceedings of the 27th International Conference on Very Large Data Bases*. VLDB '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 371–380.

Ralph Kimball and Joe Caserta. 2004. *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data* 1 edition., Wiley.

Albert Maier, Martin Oberhofer, and Thomas Schwarz. 2012. Industrializing Data Integration Projects using a Metadata Driven Assembly Line. *It - Inf. Technol.* 54, 3 (May 2012), 114–122. DOI:http://dx.doi.org/10.1524/itit.2012.0671

Jasna Rodic and Mirta Baranovic. 2009. Generating data quality rules and integration into ETL process. In ACM Press, 65. DOI:http://dx.doi.org/10.1145/1651291.1651303

A. Simitsis, P. Vassiliadis, and T. Sellis. 2005. Optimizing ETL Processes in Data Warehouses. In *21st International Conference on Data Engineering, 2005. ICDE 2005. Proceedings*. 564–575. DOI:http://dx.doi.org/10.1109/ICDE.2005.103

A. Simitsis, K. Wilkinson, U. Dayal, and M. Castellanos. 2010. Optimizing ETL workflows for fault-tolerance. In *2010 IEEE 26th International Conference on Data Engineering (ICDE)*. 385–396. DOI:http://dx.doi.org/10.1109/ICDE.2010.5447816

K. Smith, L. Seligman, and V. Swarup. 2008. Everybody Share: The Challenge of Data-Sharing Systems. *Computer* 41, 9 (September 2008), 54–61. DOI:http://dx.doi.org/10.1109/MC.2008.387

Christian Thomsen and Torben Bach Pedersen. 2005. A Survey of Open Source Tools for Business Intelligence. In A.Min Tjoa & Juan Trujillo, eds. *Data Warehousing and Knowledge Discovery*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 74–84.

E. Thoo and L. Randall. 2015. *Gartner research report: Magic Quadrant for Data Integration Tools*,

Panos Vassiliadis. 2009. A survey of Extract–transform–Load technology. *Int. J. Data Warehous. Min. IJDWM* 5, 3 (2009), 1–27.

R.Y. Wang and D.M. Strong. 1996. Beyond Accuracy: What Data Quality Means to Data Consumers. *J. Manag. Inf. Syst.* 12, 4 (1996), 5–34.

Robert K. Yin. 2009. *Case Study Research: Design and Methods*, SAGE.