

Metadata Exploitation in Large-scale Data Migration Projects

Ram Narayanan

Information Management, IBM, Southfield, MI, United States., rnarayanan@us.ibm.com

Martin Oberhofer

Information Management, IBM, Southfield, MI, United States., martino@de.ibm.com

Sushain Pandit

Information Management, IBM, Southfield, MI, United States., sushain.pandit@us.ibm.com

Follow this and additional works at: <http://aisel.aisnet.org/amcis2012>

Recommended Citation

Narayanan, Ram; Oberhofer, Martin; and Pandit, Sushain, "Metadata Exploitation in Large-scale Data Migration Projects" (2012).
AMCIS 2012 Proceedings. 6.

<http://aisel.aisnet.org/amcis2012/proceedings/DataInfoQuality/6>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISEL). It has been accepted for inclusion in AMCIS 2012 Proceedings by an authorized administrator of AIS Electronic Library (AISEL). For more information, please contact elibrary@aisnet.org.

Metadata Exploitation in Large-scale Data Migration Projects

Ram Narayanan

IBM

rnarayanan@us.ibm.com

Martin Oberhofer

IBM

martino@de.ibm.com

Sushain Pandit

IBM

sushain.pandit@us.ibm.com

ABSTRACT

The inherent complexity of large-scale information integration efforts has led to the proliferation of numerous metadata capabilities to improve upon project management, quality control and governance. In this paper, we utilise complex information integration projects in the context of SAP application consolidation to analyse several new metadata capabilities, which enable improved governance and control of data quality. Further, by investigating certain unaddressed aspects around these capabilities, often tending to negatively impact information integration projects, we identify key focus areas for shaping future industrial and academic research efforts.

Keywords

Data integration, metadata, industry experience

INTRODUCTION

Large-scale information integration projects supporting SAP implementations, Master Data Management (MDM) or Business Intelligence (BI) programs often tend to be complex enough to the extent of being difficult to manage and control. However, in recent years, advancements in commercial enterprise-scale information integration platforms (see Godinez, Hechler, Koenig, Lockwood, Oberhofer and Schroeck, 2010) have enabled the delivery of numerous *metadata* capabilities to improve project governance. That being said, managing metadata holistically with a flexible infrastructure allowing 24x7 operations with the ability to react to changes in the model used to manage metadata over its lifecycle, is not straightforward (Cao, Colgrave, Liu, Pan, Santos, Schloss, Shank, Wang, Yang, Xie 2008). Metadata, typically defined as data about data, can be primarily classified based on the type of users (business or technical) and/or the creation scope (design time or run-time) as described in Table 1. The first classification helps discern between the management and presentation of business as well as technical metadata as per the user requirement, while the second one enables a clear separation between design and run-time oriented metadata. A more comprehensive discussion on these classifications is provided in Fryman and Inmon, 2007 and Godinez, et. al. 2010. Process metadata defines the characteristics of business processes and dependencies to other metadata types, thus, in a way, tying together various different metadata. In addition, quality metadata is implicitly defined (and enabled) through dependencies on other metadata since it often requires information above and beyond what the underlying data would entail. We describe this in detail through an example later.

Metadata Type	Examples
Process Metadata	Business data requirements (BDR), process name, process dependencies and decomposition, roles and responsibilities.
Business Metadata	Business terms, business term history, business rules, authority and governance information, organizations, data provenance, etc.
Technical Metadata	Logical and physical data models, job metadata, data quality rules, etc.
Operational Metadata	Number or rows processed, data quality reports, etc.

Table 1: Types of metadata

In practice, these core classifications are often defined orthogonally or with overlaps. For instance, both technical and operational metadata have technical users as primary audience, however, technical metadata comprises design-oriented technical metadata while operational metadata comprises runtime-oriented technical metadata. Similarly, the runtime-oriented metadata is often also used for auditing purposes. Although there are various other criteria that are often used to classify metadata (e.g., structured vs. unstructured), the ones described here suffice for purposes of this discussion, where we use complex information integration projects in the context of SAP application consolidation initiatives as an illustrative example, to achieve the following two objectives: (i) firstly, we analyze how the new metadata capabilities enable improvement of the governance in a data integration project as well as the control of data quality, (ii) secondly, we identify and investigate the gaps in leveraging metadata capabilities in order to identify key focus areas to define and align future industrial and academic research efforts.

SAP APPLICATION CONSOLIDATION

For many companies, SAP Enterprise Resource Planning (ERP) systems are mission-critical and integrate internal and external management information across an entire organization, encompassing finance/accounting, manufacturing, sales and service, customer relationship management, etc. SAP systems automate this activity through an integrated software application stack. Just as important as those applications, is the data that they use to exchange and deliver the information between all business functions inside the boundaries of the organization and manage the connections to outside stakeholders.

New SAP systems require moving a high volume and variety of data in a wide array of formats from a spectrum of legacy systems that span multiple business functions. Data cleansing and conversion are critically important and a substantial undertaking. The success of the entire initiative can depend on the ability of the organization to reliably deliver high-quality data to feed and inform the new system. The ability to deliver high-quality data into a SAP target system requires an understanding of existing data quality problems and how they can be detected (a comprehensive classification of data quality issues can be found in Borek, Woodall, Oberhofer and Parlikad, 2011). The relationship of data quality issues and how they appear during data integration can be studied in Leser and Naumann, 2007. In context of this discussion, few examples of data quality issues include incorrect or incomplete address information in an SAP order to cash process, causing returned deliveries and thus, increasing operational cost. Similarly, during system integration test or production load to SAP, records might be rejected due to incorrect domain values. Data conversion issues include legacy data that is insufficient for SAP and issues detected late in the project life cycle leading to unplanned overheads and extra effort.

Metadata is the glue linking data quality issues discovered by data profiling techniques or the results of data cleansing to business entities appearing in business processes. Thus, without using metadata holistically, it is extremely difficult to govern SAP application consolidation projects (or any other major data integration project) efficiently. Using metadata to design quality processes initially, reduces the number of potential changes later. Focusing efforts to assess and understand the data priorities and accuracy of the data early in the design phase, increases the data loads and data quality and, in our experience, leads to the highest degree of accuracy in SAP application consolidation projects enabling a high ROI on the SAP implementation. A SAP application consolidation project is divided into five distinct *phases* (Figure 1), viz., Project Preparation, Business Blueprint, Realization, Final Prep and Go-live and Sustain. Further, there are four objectives, viz., Metadata Strategy, Process Design, Technical Design and Execution. Each objective can span multiple phases. Finally, each objective can be divided into one or more *tasks* such as Term Definitions, Profile, etc (Figure 2).

The **Project Preparation** phase initiates the project by means of establishing the project team, performing project planning and scoping, setting up the infrastructure and finalization of project templates. The **Blueprint** phase involves identifying the business processes required along with defining the data needs of the enterprise and designing the master blueprints to meet those needs and to ensure the fitness and quality of data for realization phase. The **Realization** phase involves planning, development, and implementation along with control activities to align the legacy data to the required business processes. This phase ensures the highest levels of data quality and consistency for data conversions to SAP. The **Final Prep and Go-live** phase involves execution of cutover steps, data conversion plan, timings, support readiness to ensure that all cutover activities can be executed within the time available and in the planned sequence, with the expected results. The **Sustain** phase is the final phase of the SAP implementation lifecycle. The purpose of this phase is to ensure that a framework and governance is in place to sustain and improve the performance of the SAP system and the data quality beyond go-live through continuous monitoring. Note that a SAP application consolidation project is usually divided into multiple deployment phases where each phase follows this five-phase structure.

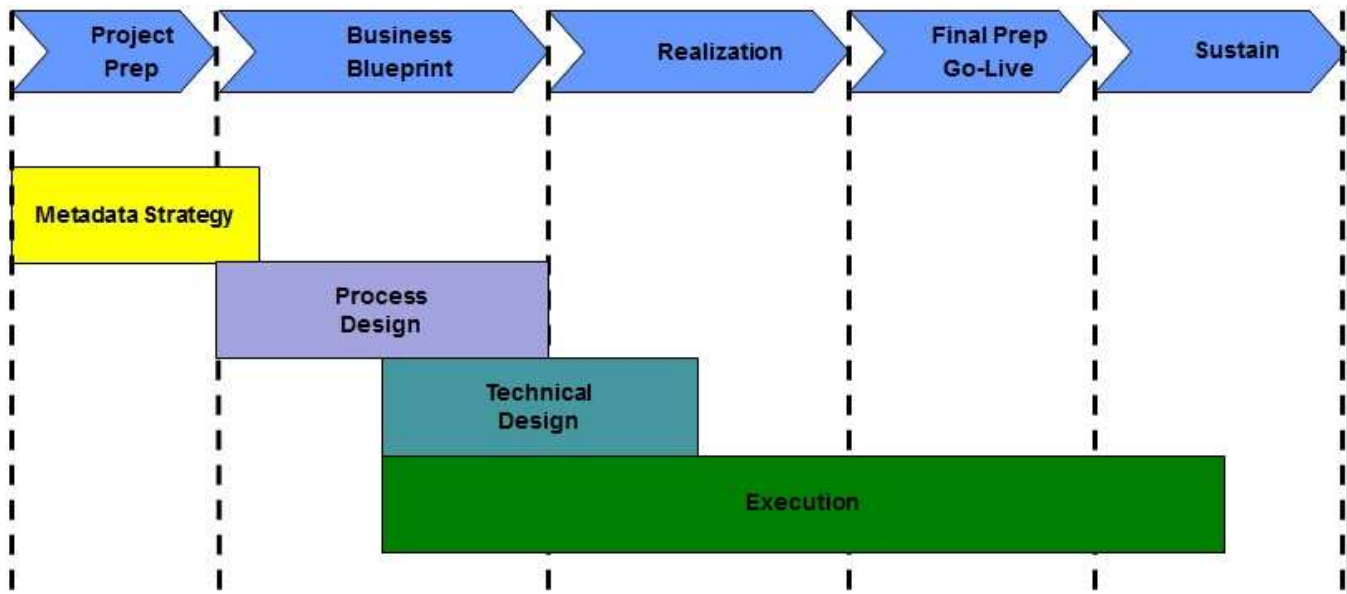


Figure 1: The project phases of an SAP consolidation project lifecycle

We now introduce the major objectives for the data team delivering the information integration in a SAP consolidation project (Figure 2):

The first objective is to define the **Metadata Strategy**. For achieving this, the organization's future enterprise metadata architecture is defined. The underlying task also recommends a logical progression of phased implementation steps that will enable the organization to realize the future vision. Business objectives drive the metadata strategy, which defines the technology and processes required to meet these objectives. The result of this process is a list of implementation phases driven by business objectives and prioritized by the business value and degree of required effort.

The second objective is the **Process Design**, where the Architecture and Data process is further refined where the users are able to clearly identify tasks, design decisions, and design intricacies. As an initial task, the data stewards identify **Term Definitions** to create the business terms. This may also include descriptions, ownership, category, synonyms, term-to-term relationships and defining valid values. Another task is the **Business Data Roadmap (BDR)** which involves the process of collecting business data requirements for each attribute and mapping these requirements to the SAP process levels. Essentially establishes a matrix of all the attributes linked to different levels as SAP business processes and serves as an input to the source-to-target mappings. And finally, the tasks of **Data Quality KPIs and Metrics** are defined for the measurement of the business critical data attributes within the context of “fitness for use”. This enables the reporting of meaningful metrics and KPIs around the defined data quality dimensions.

The third objective is the **Technical Design**, where the output of the Process Design is used to in the task **Discover / Design Logical and Physical Data Models** to identify the required structures in SAP from the Business Data Roadmap (BDR) process and to create a required logical and physical data. These models along with Business Data Roadmap (BDR), data profiling and data quality metrics provide input to the task **Source-To-Target mappings** to define the transformation rules between the source and target attributes. And finally, the task of **Job Design** serves as technical design documentation for the development of the transformation rules and data loads to SAP.

The fourth objective is the **Execution**, where the design documents are used in the development process and starts with the task **Extract**, where the initial data sources are identified for the new business process and gap analysis of the sources. Once the sources are identified, the data is extracted and landed into staging areas for further data profiling in the **Profile** task. This includes performing data analysis on the staged data – this step eliminates surprises in the source data and sets realistic expectations on the data strength and weaknesses in the legacy system. This task also helps to identify business relevance and business readiness of the legacy master data with regard to migration to SAP and provides inputs to the business rules of the data conversion. The source attributes and the data are then aligned to a common format through **Structural and Semantic Alignment**, which allows de-coupling the sources and target. This minimizes the re-work when the sources change and helps in the reuse of the target specific business rules over multiple rollouts. All the business rules are applied through **Cleanse**,

Transform and Harmonize task, which involves working with the business teams and the legacy systems SMEs to identify the data required for clean-up, enterprise standardization and recommended approach. This process involves applying the business rules to the legacy data to align to the SAP business processes and identify matched records across or within sources to be able to survive the best possible record sets for target. The transformed data goes through a **Pre-validation** task, ensuring a predictive analysis on the data gaps (difference between legacy source data and SAP requirement), where a series of re-usable gap rules are executed on the transformed data before loading to SAP. This ensures that the data will load correctly into SAP, minimizing data re-runs during the go-live period. And finally, highest quality data is loaded into SAP through **Load** task. The loading of data is based on the best practice and method for the type and amount of data being loaded.

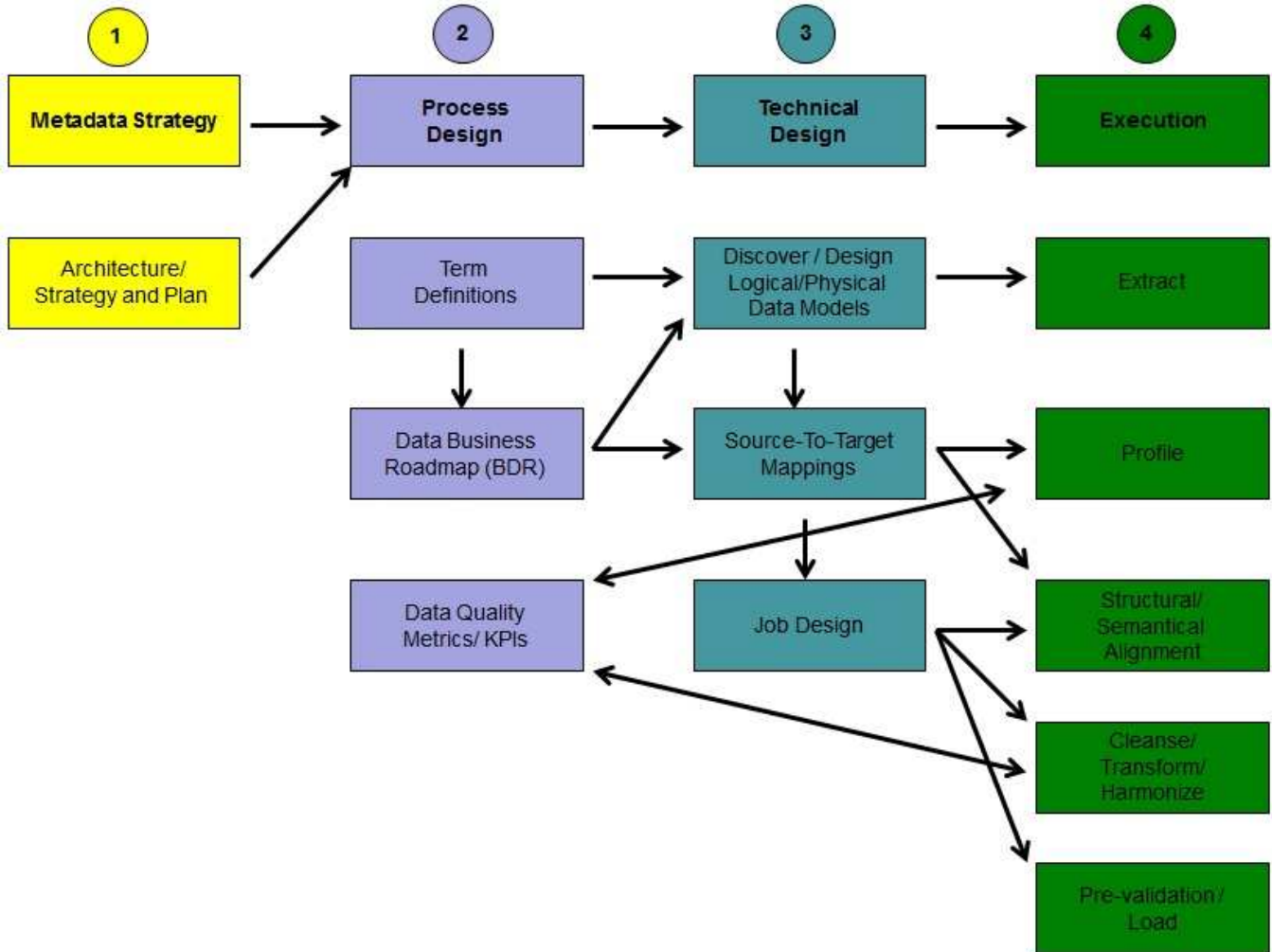


Figure 2: The major key steps in an SAP consolidation project lifecycle

Along the end-to-end process, there are checks to perform error handling and reconciliations ensuring that data leaving legacy systems reconciles with data loaded to SAP, including record counts, hash totals, financial balances, and all other required validations.

In contrast to previous approaches, a key differentiator here is the use of metadata comprehensively during the Business Blueprint phase. The created process, business, technical and operational metadata is linked end to end as part of the Process and Technical Design objectives, with specific focus on data quality. For example, with the Profile task, the data quality issues in the source systems are identified and systematically linked using Source-To-Target mappings to technical metadata (such as physical and logical data models) of the SAP target. Since the technical metadata of the SAP target is systematically linked to business objects and business processes in SAP through BDR, the data quality issues are understood in terms of the business and can be meaningfully presented to business users through relevant Data Quality Metrics and KPIs. Continuing

the SAP application consolidation scenario, we describe corresponding metadata management capabilities in Table 2. Business and (part of) process metadata are usually created while the business requirements and business processes for a new target system are being defined as part of the business design. Ideally, systematically captured business metadata tends to be the input for technical design phase. In our SAP application consolidation scenario, business terms, business data requirements, strategy objectives, and so on, fall under this class. In addition, data quality key performance indicators (KPI) are defined with respect to the context enabled through other business metadata defined in this phase. An address standardization quality metric on a database column, would only be possible if there is existing metadata capturing the fact that the column values correspond to addresses (divided into house number, street, city, zip code, etc.).

SAP Project Phase	Objective	Metadata Type	Task using Metadata	Example	Tool Example	Role
Project Prep	Metadata Strategy	Process Metadata	Define business objectives, required technology & processes	Metadata strategy objectives		Business User
Business Blueprint	Process Design	Business Metadata	Define & Maintain Business Term Definitions	Business Terms	Business Glossary	Business User, Data Steward
		Process Metadata	Business data roadmap	Business data requirements	BDR tool	Business User, Requirements Analyst
		Business Metadata	Define and Maintain Data Quality KPIs	Address Standardization Requirement	Business Glossary	Business User, Data Steward
Business Blueprint & Realization	Technical Design	Technical Metadata	Discover, Design & Maintain Logical Data Models	Logical Data Model	Data Modeling Tool	Functional Data Analyst, Data Architect, Data Steward
		Technical Metadata	Discover, Design & Maintain Physical Data Models	Physical Data Models, Table Definitions	Data Modeling Tool	Data Architect, Database Administrator
		Technical Metadata	Define Data Mappings Between Two Systems	Source-To-Target Specification	ETL Tool	Functional Data Analyst, ETL Developer
		Technical Metadata	Design & Maintain ETL Jobs Using Metadata	Extract Job	ETL Tool	ETL Developer
		Technical Metadata	Define Data Profiling	Data Profiling Job	Data Profiling Tool	ETL Developer
		Technical Metadata	Design and Maintain Cleansing, Matching and Survivorship Rules	Cleansing and Standardization Job, Address Validation Job, Matching Job, Survivorship Job	ETL Tool	ETL Developer
		Technical Metadata	Design & Maintain ETL Jobs Using Metadata	Transform Job, pre-validation and Load Job	ETL Tool	ETL Developer
		Technical Metadata	Data Lineage	ETL Jobs	ETL Tool	ETL Developer
		Technical Metadata	Impact Analysis	ETL Jobs	ETL Tool	ETL Developer, Project Manager
Realization, Final Prep, go-live & Sustain	Execution	Operational Metadata	Create and Report Runtime Statistics	ETL Jobs, Data Profiling Jobs	ETL Tool, Data Profiling Tool	Creation happens automatically, Reports consumed by ETL Developers
		Operational Metadata	Data Quality Reports	Data Profiling Jobs, ETL Jobs	Data Profiling Tool, ETL Tool	ETL Developer, translated for Business User and Data Steward

Table 2: Metadata Management Capabilities

Next, the technical design phase produces artifacts that define what needs to be built for the ETL phase. In our SAP scenario, this translates to discovery, design and maintenance of logical and physical data models, ETL job creation, profiling, data standardization, matching and de-duplication, etc. as shown in Table 2 (all these tasks are defined in depth in Godinez et. al. 2010). Further, using data lineage and impact analysis, changes during the technical design phase can be seamlessly analysed for their impact. Finally, the execution phase produces operational metadata if data quality KPIs are measured or ETL jobs are executed, whereby the results are displayed as reports to users (e.g., ETL developers). In addition, operational metadata is also produced while profiling information to understand its compliance with quality KPIs.

GOVERNANCE AND METADATA IN THE PROJECT

In this section, we illustrate the use of metadata at various stages of data integration projects. We show through examples that the governance of complex information integration in projects such as SAP application consolidation is non-trivial and often relies on human-based control management procedures. We also argue that these issues arise due to the lack of appropriate metadata capabilities.

Metadata during process design

In the process design, business term definitions and the functional data requirements of the SAP target system are captured using the Business Data Roadmap (BDR) during the Blueprint phase. Linking technical metadata of fields and tables to business objects, which are in turn linked to the process metadata, helps in fine-granular decomposition of the business processes. Here, the BDR proves beneficial because conflicting data requirements can be identified across fine-granular decompositions of business processes, substantially improving the quality of the data design. In addition, the business owner for each field is captured at the process level improving various governance aspects. While these advancements improve the situation substantially leading to better estimations of work efforts for the Realization phase, few more enhancements are possible. Firstly, for packaged applications, it would be ideal if a post-installation mechanism would exist in order to discover business process hierarchies from coarse-grained to fine-grained process levels and automatically fetch their associated business objects including the logical table and field definitions. This would avoid the manual effort required to create and maintain the BDR tool, which tends to be cumbersome and time-intensive and requires updates for each new release of a packaged application.

Also, mechanisms that are automatically able to link business terms from an enterprise business glossary and/or enterprise ontologies (see Dietz, 2006 for an in-depth discussion on enterprise ontologies) to process metadata and/or related technical metadata that is describing the logical data model of the target system, would be highly beneficial and would substantially reduce the manual labour required for the integration. Even a semi-automatic procedure making suggestions would improve upon the state of the situation.

Versioning

Code versioning is a well-understood task in many development processes across many programming languages. In data modeling and data integration domains, versioning of metadata is partially supported. Using modeling tools to create, import, change and maintain logical and physical data models over time by versioning them at appropriate steps in lifecycle and checking-in these metadata artefacts into appropriate versioning control systems, is well-understood and applied in many data integration projects. The challenge though, is to create a version of all metadata artifacts belonging to the same release of a data integration project. For example, logical and physical data models, related business terms in enterprise ontologies, metadata for ETL jobs, process metadata artefacts, etc. may all exist in different tools and repositories. We may then want to create a version-specific package of all these artifacts irrespective of the tool and package boundaries. Even though, within the domains of data modelling and ETL, tooling efforts have been made to provide a single metadata platform through which all the tools can share and consume metadata, the relevant metadata subset that is required for a particular application is still spread across multiple repositories with little or no interconnection. In order to ensure consistency, strong project management and change control capabilities are required to ensure that appropriate and consistently versioned packages of metadata are created across these fragmented metadata silos.

Data Lineage and Impact Analysis

As shown in Figure 3, there are three types of legacy source systems in an SAP application consolidation project. The first type comprises those legacy systems, which are sunset as soon as the data is migrated to the new SAP target system and once it goes live. The second type is one which co-exists temporarily with the new SAP target system for a while (and are usually sunset once a multi-phase deployment of the SAP target across different business units and/or geographies is completed). The

last type comprises those that are strategic – they remain live even after the deployment of the SAP target is completed. Note that in any SAP application consolidation project, there are usually multiple sources involved and data cleansing and harmonization, including data de-duplication, happens across all involved sources. Today, metadata tools provide capabilities such as data lineage and impact analysis. The scope of the current tools is focused around typical user roles in data integration projects such as information architects and ETL developers. This means in the case of data lineage, for a field in a table, the data can be traced to its originating field (or in some cases multiple fields) present in one or multiple source tables in corresponding source systems. This helps to understand the interdependencies of information between systems through ETL code and information systems. Impact analysis capabilities are usually able to provide insight into the ETL jobs and tables that are affected when a field or table is added, modified or deleted. This provides the developers with a baseline for estimating the cost of a change request.

The first major challenge for the temporary and strategic legacy systems is that once the initial implementation phase is completed, these systems usually need to exchange information with the new SAP target system. This is done either via periodic batch processes that are running on an ETL platform or through an Enterprise Service Bus (ESB), providing real-time integration interfaces. Now, on a logical data model level, the source-to-target mapping that is created during the Blueprint phase is independent from the underlying integration technologies such as ETL and ESB infrastructures. Problems arising from this are as follows:

- A logical source-to-target mapping created with an ETL tool cannot be seamlessly consumed by the development tools for ESB interfaces since the underlying metadata infrastructures are different (Friedrich, 2007).
- Data lineage and impact analysis tools available in the ETL domain today cannot be used to perform similar operations for the same logical source-to-target mappings in the ESB domain since within ESB, data models are in XML formatted messages, whereas the ETL view is generally comprised of relational models. As a result, for the second deployment phase (also for change requests during the Realization phase of the first deployment), it is not possible to seamlessly analyse the impact of a change request across the ETL and ESB infrastructures. The ETL process and the associated (near-) real-time interfaces in question share the same logical source-to-target mapping. The only difference is that the ETL interfaces are used for initial load whereas the associated (near) real-time interfaces between temporary/strategic legacy systems and the SAP target are required at the point the SAP target is live.

The second major challenge is that data lineage and impact analysis tools do not understand business process metadata. That means that they can neither express their results in terms of related business process to which the determined data lineage belongs nor determine what business processes would be affected in case an ETL flow changes. This prevents the ETL architects and ETL developer from seamlessly explaining the connections between ETL work and process and/or project impact to business users and project managers. As a consequence, it is difficult to fully understand a process design change request during Realization phase, which often negatively affects a system integration test or go-live effort.

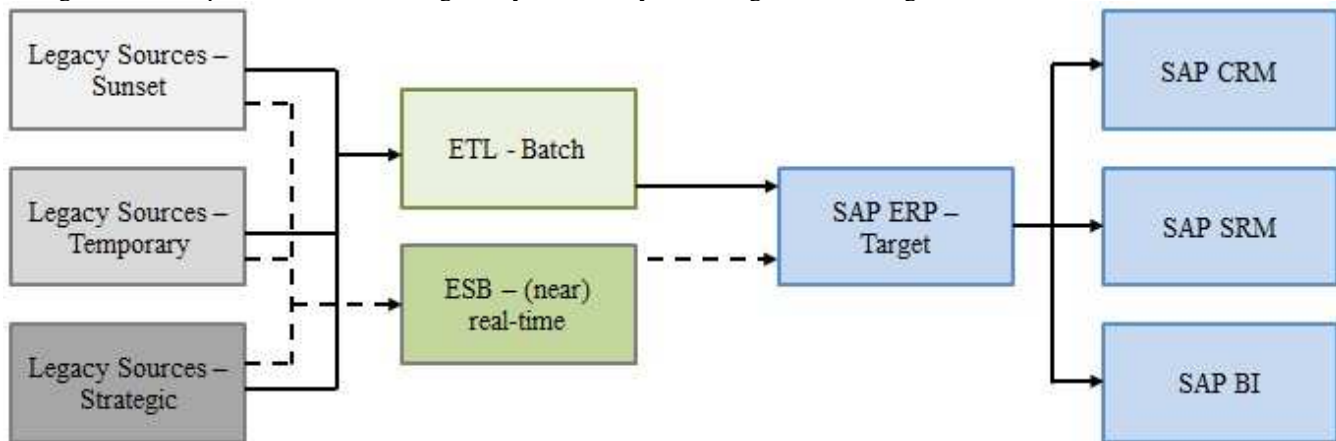


Figure 3: Migration Landscape

Finally, as shown in Figure 3, an SAP application consolidation project might involve more than just implementation of an SAP ERP system. Many companies implement one or more of the following SAP systems as well: SAP Customer Relationship Management (CRM), SAP Supplier Relationship Management (SRM) and SAP Business Intelligence (BI). SAP provides pre-built integration between the SAP applications in some cases, which results in reduced integration costs. This also implies that data might be only loaded to SAP ERP with ETL processes and pre-built integration to SAP BI is leveraged to transport the data to the SAP BI system instead of building a separate ETL batch process. While it saves cost on the integra-

tion from a development perspective, such pre-build integration creates issues from a project governance perspective since metadata tools often tend to have limited access to the metadata that is consumed by integration mechanisms built into SAP-like applications. As a consequence, data lineage and impact analysis from a report executed in SAP BI, all the way back to the legacy source systems (at least relevant for the temporary and strategic legacy systems), is not possible. This is particularly severe in cases where custom tables and attributes have been added to SAP. Also, all transformation rules embedded in pre-built integration interfaces need to be manually extracted to verify that they process the data as expected by the business users.

Metadata use to track project progress

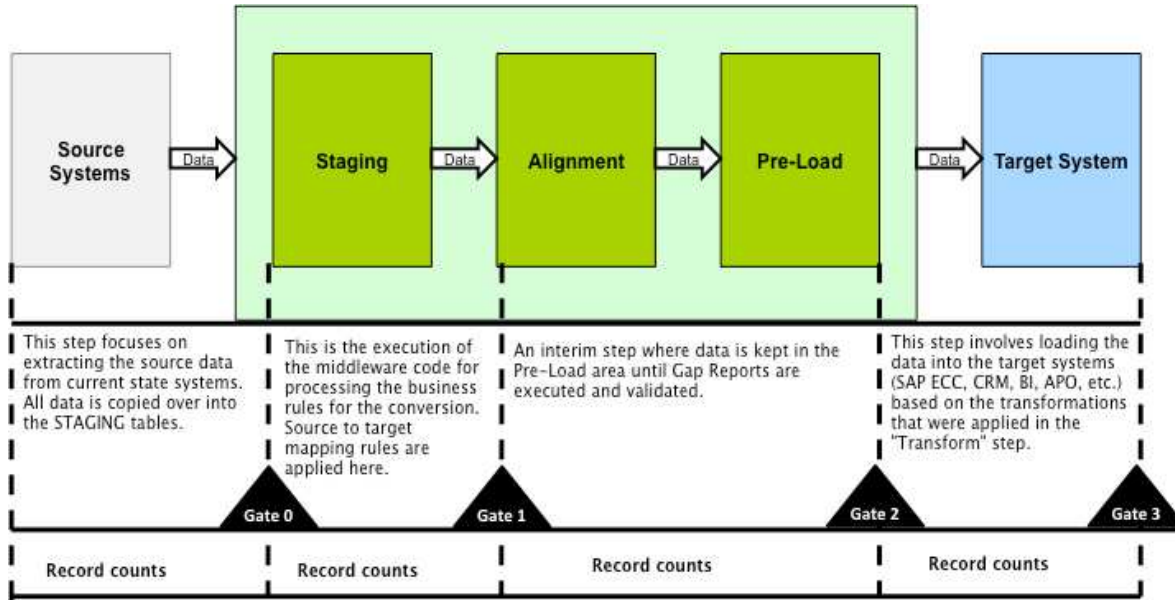


Figure 4: A metadata-based project governance report

Operational metadata is often used to track progress. For example, counts on the number of records that have been extracted from the source, the stage at which they are within the data integration development lifecycle, number of omitted records (e.g. due to an omission step in a de-duplication task), number of records that have been successfully loaded into the target system, etc. are typical statistics for progress tracking in data integration projects. These types of reports are very useful for developers and other technical resources such as information architects and are being used successfully in the field today. However, operational metadata-based reports fall short on revealing the progress of the data integration project regarding critical data quality KPIs as well as the relationship to key business process domains. Data integration project reports, measuring progress regarding data quality KPIs that are associated with key business process areas, are either not available in today’s metadata infrastructure or very costly, project-specific custom built solutions. The root cause for these issues can again be traced back to the missing link between process and data design.

Metadata and Business Rules

During a full implementation, the source-to-target business rules and metadata is captured in the design phase and used as an input to development. However, many enterprise implementations start off with a pilot implementation either by a small line-of-business or geography to test the functionality before embarking on the full-scale implementation. On such instances, where one already has a pilot implementation completed, the reverse engineering to capture the business rules and metadata from an existing environment is not possible in an automated way. Today, the teams have to manually browse through the existing business rules and metadata in spread-sheets to find gaps between the pilot and the full implementation and to leverage any existing business rules.

CONCLUSIONS AND FUTURE RESEARCH

We presented a comprehensive analysis of several new metadata capabilities by discussing complex information integration projects in the context of SAP application consolidation. By demonstrating that the governance of complex information integration is a non-trivial task that relies on manual control management procedures and that issues often arise due to the lack of appropriate metadata capabilities, we motivated a need for bridging the gap that exists due to various unexplored metadata capabilities. Opportunities for extending the present state of these capabilities exist throughout the information integration project lifecycle. During process design, although BDR provides fine-granular control for business processes by linking technical metadata to process metadata, there are still avenues for providing extra capabilities like controlling granularity while discovering business process hierarchies. Similarly, there is significant work to be done on linking (semantically or otherwise) business terms with technical and process-oriented metadata. Maintaining consistent metadata versioning presents similar interesting open problems to ensure that appropriate and consistently versioned packages of metadata are created across various different metadata silos. From lineage and impact analysis perspective, fair bit of exploration needs to be done around incorporating business process metadata in existing tooling. Overall, there is a significant amount of interesting open problems in this space for the academic as well as industrial research community to explore upon and contribute to, both in terms of fundamental as well as applied research.

References:

1. Cao, F., Colgrave, J., Liu, S., Pan, Y., Santos, C., Schloss, B., Shank, K., Wang, C., Yang, Y., Xie, G. (2008) Supporting Ontology-Based Dynamic Property and Classification in WebSphere Metadata Server. Proceedings of the 7th International Conference on The Semantic Web, pp. 861 - 874.
2. Dreibelbis, A., Hechler, E., Milman, I., Oberhofer, M., van Run, P., Wolfson, D. (2008) *Enterprise Master Data Management: An SOA Approach to Managing Core Information*, 1st Edition, Pearson.
3. Friedrich, J. (2007) *Meta-data version and configuration management in multi-vendor environments*. Proceedings of the 2005 ACM SIGMOD international conference on Management of data, pp. 799-804.
4. Fryman, L., Inmon, W.H., O'Neil, B. (2007) *Business Metadata. Capturing Enterprise Knowledge*, 1st Edition, Morgan Kaufmann.
5. Godinez, M., Hechler, E., Koenig, K., Lockwood, S., Oberhofer, M., Schroeck, M. (2010) *The Art of Enterprise Information Architecture: A Systems-Based Approach for Unlocking Business Insight*, 1st Edition, Pearson,.
6. Leser, U., Naumann, F. (2007) *Informationsintegration. Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen*, 1st Edition, dpunkt.Verlag.
7. Borek, A., Woodall P., Oberhofer, M., Parlikad, A. K. (2011) *A Classification of Data Quality Assessment Methods*. Proceedings of the 16th International Conference on Information Quality, Adelaide, Australia.
8. Dietz, J. (2006) *Enterprise Ontology. Theory and Methodology*. 1st edition, Springer.
9. Even, A., Shankaranarayanan, G., Stephanie Watts (2006) *Enhancing Decision Making with Process Metadata: Theoretical Framework, Research Tool, and Exploratory Examination*. Proceedings of the 39th Hawaii International Conference on System Sciences, pp. 209a - 209a.