



Leveraging a model-based approach for data integration analysis and design

Introduction

When it comes to information, all large enterprises share a common goal: they want to turn data into actionable business insight as quickly as possible. Businesses have a critical need to leverage information across many business channels, offerings and territories to build a unified view of their customers and business operations. Unfortunately, the complexities and constant changes from growth, mergers and acquisitions, continual IT investments and maturing software solutions and services have made it problematic to deliver an enterprise data warehouse in a minimum amount of time that is flexible enough to manage rapid changes based on business requirements.

The emergence of pre-defined, industry-specific data models have gone a long way to provide a more productive and prescribed method for gathering business requirements that lead to the design and implementation of the enterprise data warehouse model. A model-driven approach, such as that used by the IBM Industry Models, offers a framework that permits the many types of information models required by complex systems to be stored and inter-related in a consistent fashion (see Figure 1). By doing so, it provides a comprehensive blueprint of how to model data by industry, such as banking, insurance, financial markets, retail, telecommunications and health plan. For example, the IBM Banking Data Warehouse Model (BDWM) provides a logical entity-relationship model of an enterprise-wide central data warehouse.

In the same way that a prescribed industry data model can simplify and speed the implementation of the warehouse by serving as a blueprint for its construction, having a blueprint for data integration is also a critical component of a data warehouse infrastructure. Nearly 70 percent of the cost and risk of a data warehousing project occurs in the definition, design and development of data integration processes. There is a need to extend the model-driven approach into the data integration layer. With many projects, the usual method for analyzing, designing and building ETL or data integration processes is a painfully time-consuming and laborious process prone to error and misinterpretation.

New unified platforms for enterprise data integration like IBM Information Server, combine data profiling, data quality, data transformation and active metadata services to provide a framework for creating reusable data integration models. To improve the design and development efficiencies of data integration processes – in terms of time, consistency, quality and reusability – there should be a design and development technique for data integration with the same rigor used in developing data models. In short, there is a vital need for data integration models. This white paper discusses the rationale of leveraging industry-based data integration models and describes how IBM Global Business Services Industry Data Integration Models can help accelerate the population of industry data models to speed delivery and lower implementation costs and risk.

The challenges of traditional data integration methods

Typically, a data analyst charged with designing and building data integration processes documents the requirements for source-to-target mapping using Microsoft® Excel® spreadsheets, which are then given to an ETL developer for the design and development of maps, graphs and/or source code. This method of documenting requirements from source systems into Excel, and then mapping them into an ETL or data integration package has several drawbacks, including:

Lost time. It takes a considerable amount of time to copy source metadata from source systems into an Excel spreadsheet. The same source information must then be re-keyed into an ETL tool. This source information metadata captured in Excel is largely non-reusable, unless a highly manual review and maintenance process is instituted.

Non-value add analysis. Capturing source-to-target mappings with transformation requirements contains valuable navigational metadata that can be used for data lineage analysis. Capturing this information in an Excel spreadsheet does not provide a clean automated method of capturing this valuable information.

Mapping errors. Despite best efforts, manual data entry often results in incorrect entries. For example, incorrectly documenting an INT data type as a VARCHAR in an Excel spreadsheet will require valuable ETL developer time to analyze and correct.

Lack of standardization-levels of effort. Data analysts who perform the source-to-target mappings manually often capture source/transform/target requirements at different levels of completeness. When there is not a standard approach to the requirements and design of the data integration processes, there can be misinterpretation by the development staff in the coding requirements found in the Excel source-to-target mapping documents, and can result in coding errors and lost time.

Lack of standardization-inconsistent file formats. Most environments have multiple extracts in different file formats. What is needed is a move toward “read once, write many,” with consistency in extract, data quality, transformation, and load formats.

In addition to all these issues, most analysis and design work in the data integration space does not follow a common and consistent approach, with a common design format, development method and design techniques.

Moving forward, a new modeling technique that uses consistent model artifacts, similar to traditional process modeling, can and should be adopted for data integration. This technique – known as data integration modeling – is critical to achieving a higher level of quality and standardization, while simultaneously helping to minimize project risks and costs.

From process modeling to data integration modeling

Process modeling is a means of representing the inter-related processes of a system at any level of detail with a graphic network of symbols, showing data flows, data stores, data processes and data sources/destinations. These techniques are typically used to represent processes graphically for clearer understanding, communication and refinement.

There are several types of process models, including:

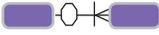
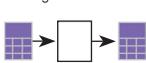
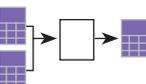
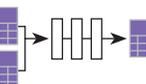
- *Process dependency diagrams*
- *Structure hierarchy charts*
- *Data flow diagrams*

Process modeling—unlike data modeling—has several different techniques based on the many different types of process interactions. Even within well-known process modeling techniques such as data flow diagramming, there are several different types of data flow diagrams, including: context diagrams, Level 0 and Level 1 diagrams and “leaf-level” diagrams.

Data integration modeling is a type of process modeling technique that is focused on engineering data integration processes into a common data integration technique. A data integration model provides a detailed representation of a data integration process for a project or business area. The development of data integration processes is similar to those in database development. In developing a database, a “blueprint,” or model of the business requirements is necessary to ensure that there is a clear understanding between parties of “what” is needed. In the case of data integration, the data integration designer and the data integration developer need that “blueprint” or project artifact to ensure that the business requirements in terms of the sources, transformations and targets that are needed to move data have been clearly communicated via a common, consistent approach. The use of a process model specifically designed for data integration can accomplish that requirement. One based on a common architectural blueprint.

Data integration modeling is a process modeling technique focused on engineering data integration processes into a common data integration architecture.

The Modeling Paradigm

	Model Type	Data	Integration	Target Audience
Less ↑ ↓ Detail ↓ More	Conceptual Models	Conceptual Data Model 	Conceptual Data Integration Model 	Data Integration Architect: Who is responsible for providing project-level data architecture planning and design expertise on development projects.
	Logical Models	Logical Data Model 	Logical Data Integration Model 	Data Integration Analyst: Crafts the logical aspects of data movement which includes, provides source-to-target mapping and transformation logic, builds Logical Data Integration models.
	Physical Models	Physical Data Model 	Physical Data Integration Model 	Data Integration Designer: Converts the Logical Model into a Physical Model in line with the technical environment, tool capabilities, and project requirements.
	Implementation Technology	Database 	Data Integration Source Code 	Data Integration Developer: Responsible for converting the Physical Model to source code and performing unit testing.
	Development Technology	Data Modeling Tool	Data Integration Package	

The structure approach for data models is relatively simple; usually there is only one logical model for a conceptual model, and there is only one physical model for a logical model. Entities may be decomposed or normalized *within* a model, whereas in process modeling, processes are decomposed into separate models.

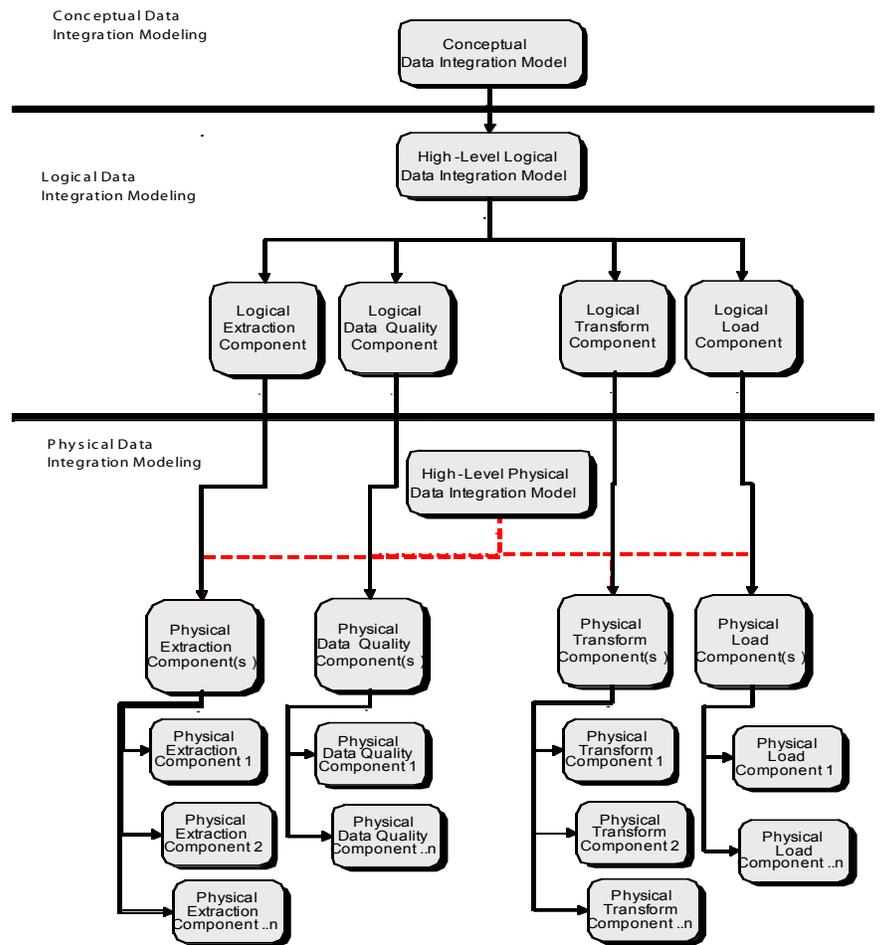
Data integration modeling follows the same approach as process modeling, where the models are broken down, or decomposed into increasingly specific models, based on the processing requirements. These include conceptual data integration models, logical data integration models and physical data integration models (see Figure 1).

The *conceptual data integration model* offers an implementation-free representation of the data integration requirements for the proposed system that will serve as a basis for scoping how they are to be satisfied and for project planning purposes in terms of source systems analysis, tasks, duration and resourcing. At this stage it is only necessary to identify the major conceptual processes in order to fully understand the users' requirements for data integration, and plan the next phase.

The *logical data integration model* helps produce a detailed representation of the data integration requirements at the dataset (entity/table)-level to detail the transformation rules and target logical datasets (entity/tables). Considered technology-independent, the focus at the logical level is on the capture of actual source tables and proposed target stores.

Finally, the *physical data integration model* is designed to produce a detailed representation of the data integration requirements at the dataset (table) level, that details the transformation rules and target physical datasets (tables). Considered technology-dependent, best practice dictates that there may be a one-to-many physical model for each logical model. The focus at the physical level is on the decomposition of the logical transformations into the data integration architecture (DIA), e.g., initial stage, cleansed stage or load-ready stage.

Structure of Data Integration Models



What is a Reference Architecture?

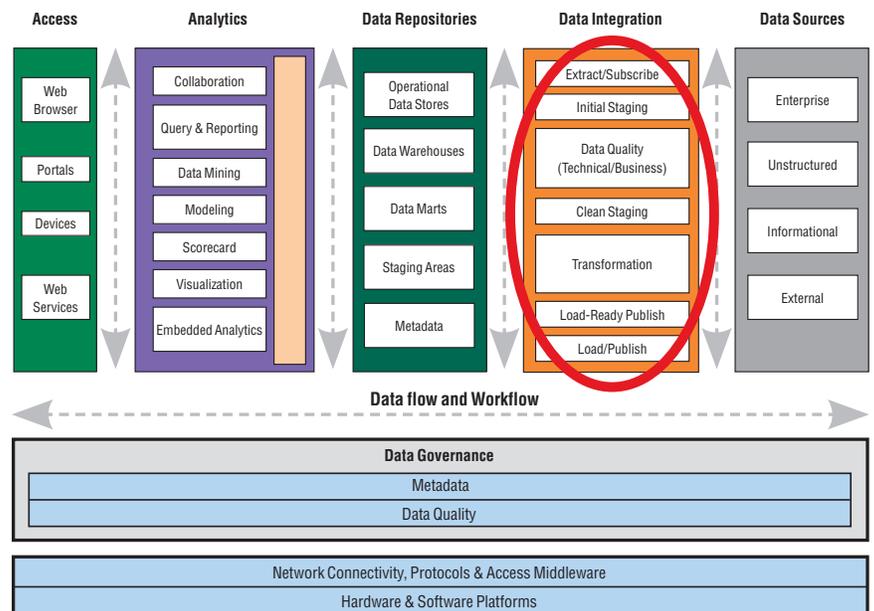
When we construct a house or building there are always certain common components, such as:

- Foundations
- Water infrastructure
- Electrical infrastructure
- Telecommunications
- Heating and cooling

Similarly, there are common components that all data warehouses share. Requirements dictate design and the Reference Architecture is the blueprint.

What is a the IBM Business Intelligence Reference Architecture?

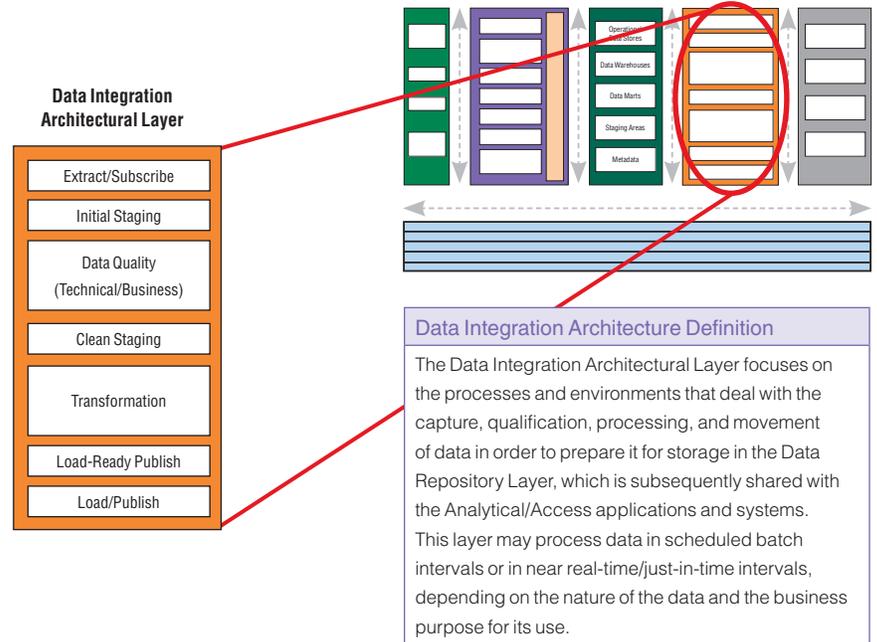
- The Business Intelligence Reference Architecture represents a component-based, scalable, conceptual architecture. Each component layer can be described in terms of the people, processes and technology that it comprises.
- The Business Intelligence Reference Architecture is sufficiently flexible to support the unique requirements of each customer’s business problem.
- The Business Intelligence Reference Architecture may itself be a component within the broader framework of an enterprise-wide technical infrastructure.



Using the IBM BI Reference Architecture as a framework, we have developed a detailed publish and subscribe architecture for the Data Integration Layer.

The IBM Data Integration Architectural Layer

Using our BI Reference Architecture, we have further defined based on engagement experience a detailed and proven Data Integration Architecture with common conceptual, logical, and physical components. These components are designed to optimize the inherent strengths of the Data Integration technologies.



Following this architectural pattern leads to a common and consistent process for loading the data warehouse.

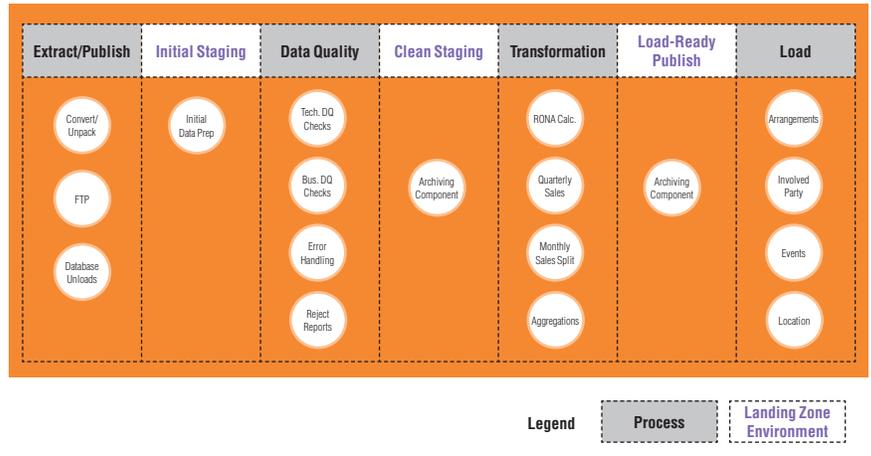


Figure 1: Illustrates how conceptual, logical, and physical graphs are broken down.

Mapping data integration modeling to the data integration architecture

Data integration modeling defines the process of integrating data based on a blueprint or architecture. For data integration, there is a defined architectural framework or data integration architecture. The data integration architecture focuses on the methods and constructs that deal with the processing and movement of data to prepare it for storage in the operational data stores, data warehouses, data marts and other databases in order to share it with the analytical/access applications and systems. This architecture may process data in scheduled batch intervals or in near real-time/just-in-time intervals, depending on the nature of the data and the business purpose for its use. Using the data integration architecture as a framework enables organizations to model the discrete processes and constructs (see Figure 2).

Process areas of the data integration architecture include:

- **Extract area.** *Extract/data movement is the set of tools and processes that get data from sources to an Initial Staging Area. The data integration environment provides mechanisms that will allow data to move from source system platforms to the data integration platform for further processing and transmission.*
- **Initial Staging Area.** *The area where the copy of the data from sources persists as a result of the extract/data movement process. (Data from real-time sources that is intended for real-time targets only is not passed through Extract/Data Movement and does not land in Initial Staging Area.) The major purpose for Initial Staging Area is to persist source data in non-volatile storage to achieve the “pull it once from source” goal.*
- **Data Quality Area.** *Provides for common and consistent data quality capabilities through a standard set of data quality reusable components created to manage different types of quality checking. The outputs of the data quality functions or components will link with exception handling.*
- **Calculations and Splits Area.** *The data integration architecture supports a data enrichment capability that allows for the creation of new data elements (that extend the data set), or new data sets, that are derived from the source data.*
- **Clean Staging Area.** *Contains records that have passed all data quality checks. This data may be passed to processes that build load-ready files and may also become input to join, split or calculation processes which in turn produce new data sets.*
- **Process and Enrichment Area.** *The data integration architecture supports capabilities for joins, lookups, aggregations and delta processing functions.*
- **Target Filtering Area.** *The first target-specific component to receive data. Target filters format and filter multi-use data sources from Clean Staging Area, making them load-ready for targets. Both vertical and horizontal filtering is performed.*
- **Load-Ready Staging Area.** *Utilized to store target-specific load-ready files. If a target can take a direct output from the ETL tool first without storing the data first, storing it in the Load-Ready Staging Area may not be required.*

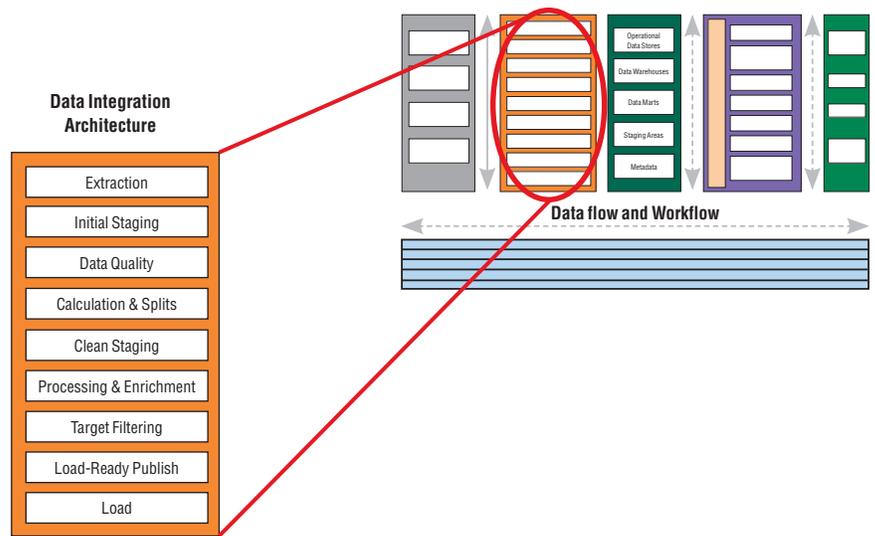


Figure 2: The data integration architecture process areas.

Using the data integration architecture as a framework, organizations can model the discrete processes and constructs (see Figure 3).

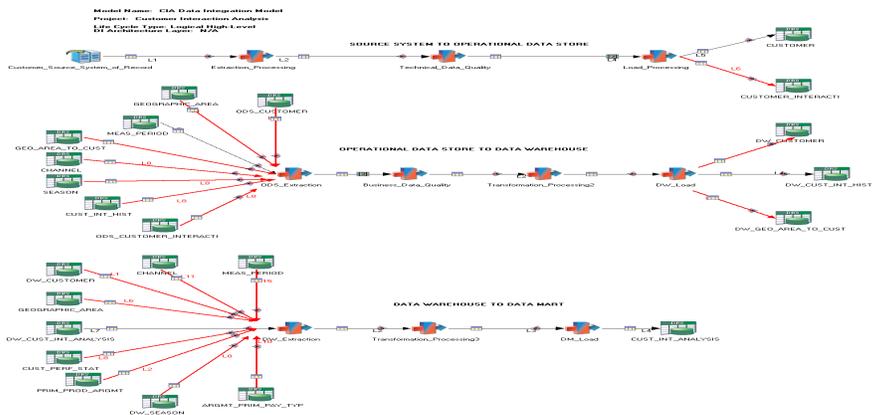


Figure 3: Modeling discrete processes and constructs within the data integration architecture framework.

Benefits of using data integration models

Using a common and consistent approach through the use of modeling data integration requirements and designs offers data integration projects several benefits, such as increased quality, standardization, metadata capture and consistency, which can also help reduce project risk and rework.

In addition, using data integration technologies and metadata to create the data integration models can help organizations further leverage the investment they've already made.

Key benefits include:

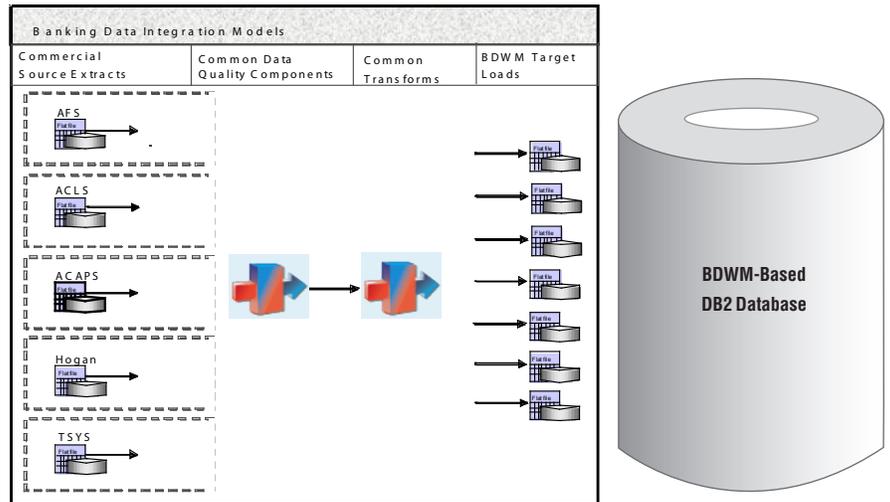
End-to-end communications. A faster transfer of requirements from data integration designer to data integration developers can provide higher quality results. By transferring a logical data integration model using common modeling techniques, organizations can automate metadata transfer, keep requirements at the same level and significantly reduce the risk of mapping errors.

Development of leverageable enterprise models. The benefits of reuse are much more easily realized when macro-design, micro-design and build-cycle components can be reviewed visually. There are many methods to accomplish the development of leverageable models, such as MS-Visio. However these tools require manual creation and maintenance to ensure that they are kept in synch with source code and Excel spreadsheets. The overhead of the maintenance often far outweighs the benefit of the manually created models. By using a data integration tool (e.g., IBM Information Server DataStage), existing models can be reviewed for potential reuse, and the maintenance performed when the model is updated. The use of reusable models helps facilitate the tracking and identification of reusable code.

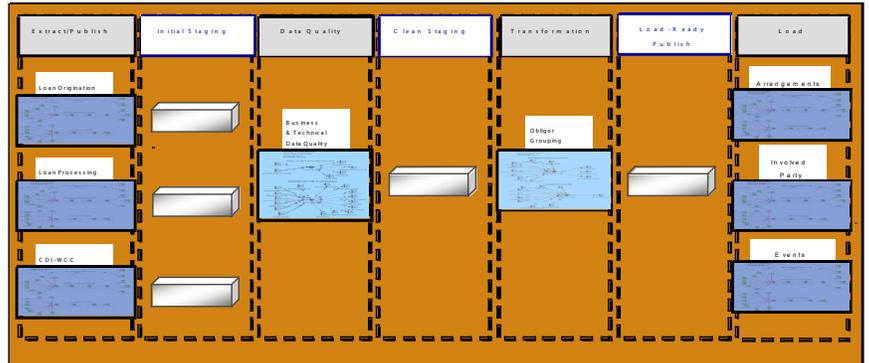
Capture of navigational metadata earlier in the process. Capturing data integration requirements as logical and physical data integration models can help organizations better leverage a data integration tool’s development environment to create the data integration processes or components that store the objects that make up the graphs directly into an enterprise metadata repository. It also provides the ability to reuse source and target data structures and transformation objects that are in the repository. The physical graphs are stored in the same repository and can be linked to each other as well as to other existing metadata objects, such as logical data models and business functions. Metadata object reuse and linking related objects gives organization the ability to perform more thorough impact analysis from a single source. The capture of source-to-target mappings with transformation requirements can be captured much earlier in the process. In addition, metadata capture will be automated, which can help decrease capture time and risk of data entry error.

IBM industry-based data integration models

We know based on our experiences that for each industry there are core processes and calculations, and aggregations. We have built a series of Banking Data Integration Models based on our knowledge of the Banking Data Warehouse Model subject area loads, common transformation, and commercial system extracts.



By leveraging these models against our Data Integration Architecture, we are able to provide a set of accelerators or data integration models using IBM Information Server DataStage. This is the architectural layer that best maps core banking functionality into the Banking Data Warehouse.



For example, how to aggregate obligors to loan obligations, and how to calculate Total Borrower Exposure.

The first set in a planned series of industry data integration models, the Banking Data Integration Models offers pre-built components for many core banking applications to accelerate the population of Information Framework-based databases, including (see Figure 4):

- **The Source Extraction Component** determines what subject areas will need to be extracted from sources such as applications, databases, flat-files and unstructured sources.
- **The Data Quality Component** identifies the data quality criteria for the scoped application. It identifies the critical data elements, the domain values and business rule ranges for the intended target and defines them as either absolute or optional data quality business rules. These business rules are then transformed into cleansing process specifications.

- **The Transform Component** identifies at a logical level what the business rules are for the target data store to determine what transformations (in terms of calculations, splits, processing and enrichment) are needed on the extracted data to meet the business intelligence requirements in terms of aggregation, calculation, and structure.
- **The Target Load Component** determines at a logical level what is needed to load the transformed and cleansed data into the data repositories.

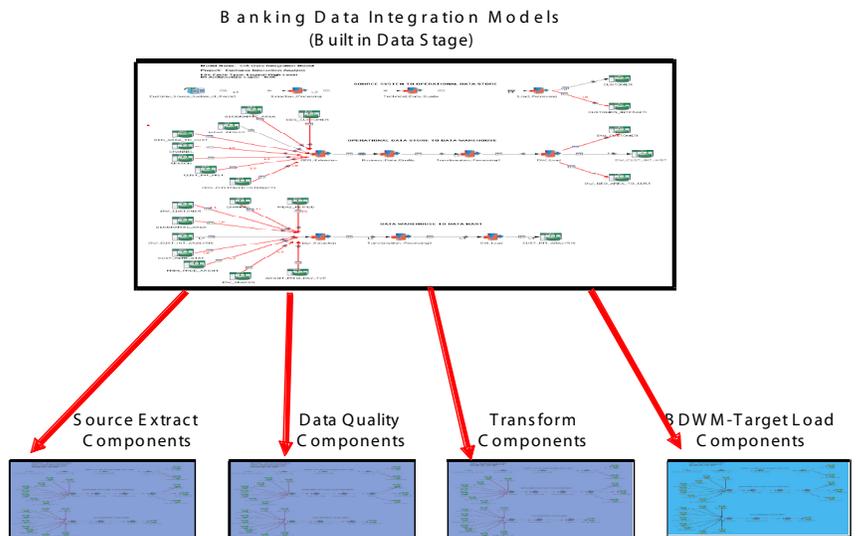


Figure 4: The Banking Data Integration Models using IBM Information Server DataStage offers pre-built components for many core banking applications.

A complete solution for enterprise data warehouses

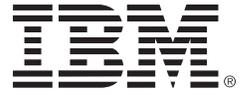
IBM is now in a unique position to offer a complete solution for enterprise data warehouses. For the data warehouse, IBM offers the IBM Balanced Configuration Unit (BCU) to provide optimal performance. Designed around the concept of a balanced infrastructure through the use of modular nodes, the BCU consists of hardware and software that IBM has integrated, preconfigured, tested and validated as a scalable solution for data warehousing systems. Using this approach, IT departments can help reduce design time, shorten deployments and maintain a favorable price/performance ratio as they add building block nodes to enlarge their data warehouses over time. To speed the requirements gathering, design, and implementation of the data warehouse model, organizations may add the appropriate IBM Industry Data Model.

IBM also offers the IBM Information Server Blade which consists of hardware and software that IBM has preconfigured, tested and validated as a scalable solution for enterprise data integration. To speed the design and implementation of the data integration processes for the data warehouse, organizations may take advantage of the appropriate Global Business Services Industry Data Integration Model. Working with IBM Global Business Services, this unique approach from IBM enables organizations to quickly design and deploy a roadmap for the enterprise data warehouse and the data integration server that ensures that both the data warehouse and the integration server scales and adapts to new business requirements over time.

Summary

Data integration modeling can bring the same type of rigor and consistency found in developing databases. By applying this rigor, organizations can better plan, design, develop and maintain the data integration processes necessary to support operational and analytic data stores. A leading provider of data warehousing systems for business intelligence, IBM has defined, tested and validated the components needed to help ensure success at virtually every stage of data warehousing.

To learn more about implementing an extended infrastructure for dynamic warehousing, contact your IBM sales representative, or visit: ibm.com/software/data/ips/solutions/ddw.html or contact your IBM representative.



© Copyright IBM Corporation 2007

IBM Software Group
Route 100
Somers, NY 10589

Produced in the United States of America
10-07
All Rights Reserved

IBM and the IBM logo are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Microsoft and Excel are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product or service names may be trademarks or service marks of others.