Building Data Warehouses Using The Enterprise Modeling Framework

Joseph O. Chan
Roosevelt University

Follow this and additional works at: https://scholarworks.lib.csusb.edu/jiim

Part of the Management Information Systems Commons

Recommended Citation
Available at: https://scholarworks.lib.csusb.edu/jiim/vol13/iss2/3

This Article is brought to you for free and open access by CSUSB ScholarWorks. It has been accepted for inclusion in Journal of International Information Management by an authorized editor of CSUSB ScholarWorks. For more information, please contact scholarworks@csusb.edu.
Building Data Warehouses Using The Enterprise Modeling Framework

Joseph O. Chan
Roosevelt University

ABSTRACT

This paper proposes an enterprise modeling framework for the deployment of data warehouses. The framework provides the information roadmap coordinating source data and different data warehouses across the business enterprise. The paper introduces a solution to address data warehousing issues at the enterprise level while avoiding the pitfalls of creating enterprise data warehouses and universal data marts. It further proposes a change of paradigm from point solutions focus to a methodology driven by enterprise requirements to meet the challenges of the new economy. The proposed framework emphasizes the separation of the conceptual construct from the physical and operational constructs of an enterprise. It points out the differences and dependencies of analytic and operational processes and how data warehouses and operational data stores respectively support their information requirements. This paper will demonstrate how the enterprise modeling framework for data warehousing can produce business benefits.

INTRODUCTION

The 90's represent the decade of technology acquisition for many companies. Confronted with the Year 2000 problem, businesses unleashed vast amounts of capital spending in upgrading old systems or replacing them with new ERP systems. However, most companies are not reaping the anticipated benefits in information access and sharing. Technologies implemented in the past decades have created the legacies confined by the boundaries of systems and organizations. Information collected in these transactional systems got buried in their respective silos and hidden from the enterprise.

As we move towards a knowledge-based economy, empowered by the real-time infrastructure of the Internet, businesses are in a hurry to gather and analyze all sorts of data for competitive advantages. For example, businesses collect data about customers to enhance customer relationship management programs. The government collects data about crimes, court dispositions and correctional methods for crime prevention. Data collection seems to have evolved into a business on its own. The challenge however is the transformation of these data into useful information for the enterprise.

Data warehousing provides the data construct to aggregate data by subject areas for the purpose of decision support. Data warehouses have certain characteristics that differentiate them from transactional databases. The classic definition by Inmon (1993) describes these characteristics as subject oriented, nonvolatile, integrated and time variant. As data warehousing technologies are made available for data extraction, transformation and loading (ETL), some use data warehousing as a means of data consolidation at the transactional level for reporting and other purposes.

A disturbing phenomenon has developed over the years, which reminds us of the issues in file processing systems a few decades ago. During that period, a specific program was created with its associated data files when a specific need arose. If there was a need to solve another problem, another program was developed and another set of data files was created. Such practices caused the data redundancy and inconsistency problems for data management across an enterprise. The database management systems that came afterwards were trying to solve the problem by putting related data in one place, at least conceptually. The current issues of data warehousing are, to an extent, similar to those in file processing. If there is a need of information to solve a problem, a data warehouse or data mart is created for the subject matter. If there is another need of information for another problem, another data warehouse or data mart is created for this other subject matter. Over time, it becomes a maintenance nightmare to coordinate and synchronize information in these data warehouses within an enterprise. Some companies resort to new strategies...
in consolidating old data warehouses into new data warehouses, and thus creating more legacy data warehouses that need coordination, synchronization and maintenance.

It is not uncommon in today's business environment that decision support requires information across the entire enterprise. Information needed to create business intelligence may require customer information, marketing and sales information, production information, supplier information ... etc. It may involve many subject oriented data warehouses in the enterprise. Furthermore, some businesses are using data warehousing as a holistic means to address both transactional and analytical data requirements. Real-time information access with unanticipated requirements may put the usage of data warehouses in the most ineffective mode. Which, if any, of the data warehouses can answer such and such queries? And what if it requires multiple data warehouses to seek the truth?

This paper discusses a logical framework that supports the information requirements both at the transactional and analytic levels for the enterprise. It points out the different levels of data structures and how they are related. Furthermore, it provides the consistency of data for data warehouses and data marts throughout the enterprise.

TWO ENDS OF THE SPECTRUM

In the evolution of building data warehouses and data marts, two practices have emerged. One advocates a centralized enterprise data warehouse that serves as the repository of data from all data sources, from which various data marts can be created. The other advocates a purely bottom-up approach where data marts are built as point solutions to departmental or functional needs isolated from the rest of the enterprise. In addressing the data warehousing issues at the enterprise level, Simon (1998) referred to these two approaches as the “big bang” and the “loose confederation” approaches respectively. Simon further pointed out that the enterprise data warehouse “big bang” approach attains the maximum data extraction and the highest degree of source-neutral information abstraction. However, as pointed out by Simon, there are many pitfalls, which include cross-organization issues, technical complexity, data semantic issues and long delivery time. On the other hand, the bottom-up approach, while providing the point solutions with respect to departmental or functional needs, will result in information silos within an enterprise over time. Companies attempt to tie together these data marts into universal data marts. While the bottom-up approach avoids some of the pitfalls of the enterprise data warehouse, it creates a different set of issues. As summarized by Simon, these issues include redundancies in data extraction from data sources, unclear objectives by integrating data marts, technical challenges in platform integration and cross-functional issues.

THE ENTERPRISE MODEL SOLUTION

Analogous to building a house, the enterprise data warehouse “big bang” approach gathers all the 2 by 4’s, drywall, plumbing and electrical units in a storage area and then tries to figure out how they will be used in the construction of each room later. In the bottom-up approach, each room is built separately sourcing its own 2 by 4’s, drywall, plumbing and electrical units. The task is then trying to fit these rooms together into a single house.

Each of these approaches focuses on the physical aspects of building a house. The enterprise data warehouse “big bang” approach puts all data in one place first, where in the bottom-up approach; each data mart sources its own data. In practice, various degrees of both of these techniques may be used depending on the requirements. For example, a garage may be prefabricated and fitted into a house easily. Drywall or floor tiles may be purchased for many rooms so that what is left over from one room can be used for another. Plumbing and electrical wiring may best follow an integrated plan so they all fit together as a unit. Therefore, a logical approach is to start with a blueprint of the house and determine a combination of these options, optimizing cost, time, and benefits. The enterprise model provides this framework in building data warehouses.

It should be pointed out that there is a fundamental difference between an enterprise data warehouse and an enterprise model. The former is a physical concept whereas the latter is a logical concept. The “big bang” approach utilizes a single data warehouse as a physical clearinghouse for all enterprise source data. It then feeds into various data marts for specialized analytical purposes. The proposed enterprise model framework differs from this approach in that the clearinghouse function for enterprise data is at the conceptual level and not at the physical level. The conceptual enterprise model serves as the information blueprint for all physical data warehouse and data mart
The physical realization of the conceptual enterprise model may consist of multiple data warehouses or data marts sourcing from one or more operational data stores or source systems.

The Enterprise Model framework is an extension of the “3-Schema” architecture originally proposed by ANSI/X3/SPARC (1975). The 3-Schema addresses the construct of data based on three levels of representation: the conceptual schema represents the logical view of data, the internal schema represents the physical data storage definitions, and the external schema represents the user application views of data. The fundamental concept is the separation of the data definition (the “what”) from its physical representations and usage (the “how”, “where”, “who” and “when”).

While the 3-Schema provides the foundation of data definition in the development of databases and their applications, the fundamental concept can be extended to enterprise information management. This extended definition of the 3-Schema will be henceforth called the Enterprise Model. Chan (2003) described the three levels of the Enterprise Model: the Conceptual Enterprise Model (CEM), the Operational Enterprise Model (OEM) and the Technical Enterprise Model (TEM). In this paper we extend this definition to include the Analytic Enterprise Model (AEM) at the same level as the OEM in the external view of the enterprise. See Figure 1 for the construct of the enterprise model framework as an extension of the 3-schema.

The Conceptual Enterprise View

Similar to the conceptual schema in the 3-Schema architecture of data, the Conceptual Enterprise Model (CEM) is a representation of the “what” in an enterprise. The CEM consists of the conceptual definition of data and business functions across business units and processes for an enterprise. The conceptual data models consist of the Enterprise Data Model (EDM) supporting the operational data requirements of an enterprise, and the Analytic Data Model (ADM) supporting the analytic data requirements of an enterprise. The Entity-Relationship modeling technique introduced by Chen (1976) can be used to develop the EDM. Dimensional modeling techniques such as the Star Schema can be used to develop the ADM. On the functional side, the Enterprise Functional Model consists of the Operational Function Model (OFM) and the Analytic Function Model (AFM). The OFM represents operational functional requirements, and the Analytic Function Model represents the analytic functional requirements of an enterprise. The analytic data and function models provide the conceptual framework for building data warehouses and analytic processes.

The External Enterprise View

Similar to the external schema in the 3-Schema architecture of data, the external enterprise view represents the usage of data describing the “how”, “where”, “who” and “when”, supporting operational and analytic requirements of an enterprise. The Operational Enterprise Model (OEM) describes the business processes, events, people and organizations that are required to implement the day-to-day business operations. For example, call center operations and Web-based self-service may be implemented at the OEM level to support customer service functions. The components of the OEM consist of the user application view, the business process view and the organization view. Similarly, the Analytic Enterprise Model (AEM) consists of different types of analytic structures for an enterprise supporting various processes in decision support, predictions, forecasts and estimations. The components of the AEM consist of the user application view, analytic process view and the organization decision view. Furthermore, there exists a feedback loop between the OEM and AEM. The OEM collects the operational data required for analytic processing in the AEM, which in turn creates the business intelligence for operations in the OEM.

The Internal Enterprise View

Similar to the internal schema in the 3-Schema architecture of data, the Technical Enterprise Model (TEM) is a representation of the physical implementation describing the technical “how”, “where” and “when”. The TEM consists of the internal views of data structure and storage, software modules, hardware platforms and communications networks. For example, the physical representation of a data warehouse can be a relational
database running on a Unix machine. The analytical processes can be deployed on the Intranet using Online Analytical Processing (OLAP) software supporting sales and marketing analysis.

The Enterprise Information Roadmap

The key concept behind the enterprise model is the physical separation of the conceptual, external and internal views of the enterprise, but yet preserving their logical connections. We will call this construct the Enterprise Information Roadmap (EIR). The EIR defines the mappings and rules of associations between the various components of the enterprise model. The virtuality of the model (the three layers and respective components being physically separated but logically connected) insulates enterprise information assets from process, organizational and technological changes. Many designer tools are available for the creation and maintenance of the mappings between these models.

Figure 1. Expansion of the 3-Schema Architecture to the Enterprise Model Framework

<table>
<thead>
<tr>
<th>The 3-Schema Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>External Schema</strong></td>
</tr>
<tr>
<td><strong>Conceptual Schema</strong></td>
</tr>
<tr>
<td><strong>Internal Schema</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The Enterprise Model Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>External Enterprise View</strong></td>
</tr>
<tr>
<td><strong>OEM (Operational Enterprise Model)</strong></td>
</tr>
<tr>
<td>User Application View</td>
</tr>
<tr>
<td>Organization Structures</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Conceptual Enterprise View</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CEM (Conceptual Enterprise Model)</strong></td>
</tr>
<tr>
<td>Enterprise Data Model</td>
</tr>
<tr>
<td><strong>OFM (Operational Function Model)</strong></td>
</tr>
<tr>
<td>Operational Function Model</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Internal Enterprise View</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TEM (Technical Enterprise Model)</strong></td>
</tr>
<tr>
<td>Physical Data Storage &amp; Structure</td>
</tr>
</tbody>
</table>

**EIR (Enterprise Information Roadmap)**

= \{RELATIONS between the components of OEM, AEM, CEM & TEM\}
BUILDING DATA WAREHOUSES

The Analytics and Operations Feedback Loop

Supporting analytic processing is the ultimate objective of building data warehouses. In the context of the Enterprise Model, the analytic processes in the AEM are at the same level as the Operational Enterprise Model (OEM). Figure 2 depicts the relationship between the AEM and OEM. Results of analytic processes affect operations, and conversely, data obtained in operations affect future analytic processes. For example, customer-centric analytic processes create customer insight, which affects the operations of a one-to-one marketing campaign. Data obtained from the one-to-one marketing campaign provides the feedback to the analytic processes, which in turn creates additional and new customer insight. A number of techniques can be used to facilitate the feedback loop between analytics and operations. They include models using statistical methods and machine learning mechanisms. Statistical methods are used to collect, analyze, and interpret data. Popular techniques in regression analysis can be deployed in statistical analysis. However, statistical methods are not useful in the discovery of complex patterns and rules in large quantities of data. Machine learning involves adaptive mechanisms that enable computers to learn from experience, examples and analogies. Popular approaches to machine learning include artificial neural networks and genetic algorithms. See Negnevitsky (2002) for a detailed description of these approaches.

Figure 2. Analytics and Operations

ANALYTIC DATA AND FUNCTION MODELS

The Analytic Function Model (AFM) describes the functionality of analytic business requirements. For example, for a retail store to determine the physical layout of products to promote cross-selling opportunities, it requires the knowledge of the association of products in transactions over a period of time and over various samples. Such requirements provide the framework for the design and selection of analytic methods. The analytic functional requirements together with the data framework provided by the Enterprise Data Model (EDM) form the basis for the design of the Analytic Data Model (ADM). The ADM associates data along subject areas supporting various dimensions for analysis. Dimensional modeling techniques such as Star Schema and Snowflake Schema can be employed to develop the Analytic Data Model as described in Todman (2001). The analytic data and function models, in conjunction with the target data warehousing and analytic processing technologies, provide the basis to the design of the respective data warehouses and analytic processes.

The Physical Data Warehouse

The physical data warehouses and all source systems belong to the Technical Enterprise Model (TEM). The Enterprise Data Model (EDM) can be physically realized as one or more Operational Data Stores (ODS). The ODS provides data consolidation of source systems. Data are extracted, transformed and loaded into the ODS from different source systems which include the legacy systems, transactional systems and external systems. It should be observed that the total consolidation of data into one data store might not be practical for large enterprises.

ODS may be of great importance in situations where multiple source systems are needed for certain data access requirements. It provides a physical representation of consolidated and transformed data from source
systems, and acts as the data feed to target data warehouses. Moreover, the ODS, not the data warehouse, serves the need for transactional reporting requirements.

The key point of the enterprise modeling approach is that the design allows each data warehouse or data mart to source its data from zero to one or more ODS. In the case of zero ODS, the data warehouse or data mart sources data directly from the source systems. This is similar to the case of the previously discussed bottom-up approach. The difference is that the sourcing of data under the enterprise model framework is guided by the enterprise information roadmap, which provides the conceptual integration of data from source systems to various target data warehouses. When all data warehouses and data marts source from one big ODS, the architecture is similar to the enterprise data warehouse big bang approach. The optimal design of the number of ODS for an enterprise will take into consideration the number and size of data sources, and their relationships to the target data warehouses.

The Enterprise Model Architecture for Data Warehouses

As illustrated in Figure 1, the enterprise model framework consists of the external enterprise view, the conceptual enterprise view and the internal enterprise view. The external enterprise view consists of the Operational Enterprise Model (OEM) and the Analytic Enterprise Model (AEM). The conceptual enterprise view as described by the Conceptual Enterprise Model (CEM) consists of the respective data and functional components: the Enterprise Data Model (EDM), the Analytic Data Model (ADM), the Operational Function Model (OFM) and the Analytic Function Model (AFM). The internal enterprise view as described by the Technical Enterprise Model (TEM) consists of the database, software, hardware and communications components. The Enterprise Information Roadmap (EIR) is the meta-model of the enterprise model. It consists of the definitions, mappings and rules of associations for all the model components of the enterprise model.

Figure 3 provides the architectural framework for building data warehouses and analytical processes based on the construct of the enterprise model. The conceptual construct of the enterprise operational data and functions are represented respectively by the Enterprise Data Model (EDM) and the Operational Function Model (OFM) in the Conceptual Enterprise Model (CEM). The conceptual construct of the enterprise analytic data and functions are represented respectively by the Analytic Data Model (ADM) and the Analytic Function Model (AFM) in the Conceptual Enterprise Model (CEM). The ADM can be constructed from the EDM by associating data entities along subject areas based on the requirements defined in the AFM. The enterprise data and operational function models provide conceptual framework for the design of the operational data stores. Similarly, the analytic data and function models provide the conceptual framework for the design of the target data warehouses. The operational data stores provide the data feeds for operational processes and reporting in the OEM whereas, the data warehouses and data marts provide the data feeds for the analytic processes and reporting in the AEM. The source data, which include the legacy, transactional and external data in the Technical Enterprise Model (TEM) are mapped to the enterprise data model (EDM). Data warehouses and data marts can source data from the operational data stores or directly from the source systems. Mappings between all components of the enterprise model are constructed through the enterprise information roadmap. The architecture also illustrates the creation of business intelligence for an enterprise through the feedback loop between analytics and operations.
Figure 3. The Enterprise Model Architecture for Data Warehouses

AEM (Analytic Enterprise Model)  OEM (Operational Enterprise Model)

Analytic Processes and Reporting  Operational Processes and Reporting

CEM (Conceptual Enterprise Model)

ADM
Analytic Data Model

EDM
Enterprise Data Model

EFM
Enterprise Function Model

AFM
Analytic Function Model

OFM
Operational Function Model

Data Warehouses & Data Marts

ODS
Operational Data Store(s)

Legacy
Data

Transactional
Data

External
Data

ETL

Published by CSUSB ScholarWorks, 2004
BENEFITS OF THE ENTERPRISE MODELING APPROACH

Separation of Conceptual, Physical, Operational and Analytical Models

While the monolithic enterprise data warehouse approach and the bottom-up data mart approach provide two extreme views, an optimal data sourcing strategy can be determined based on business and technical requirements. The enterprise model serves as the blueprint governing the relationships between source systems and target data warehouses in the enterprise. The Y2K problem has demonstrated the importance of the 3-schema in data management. The core issue lies in the embedded definitions of the two-digit year code in the application programs. Changing the two-digit code to a four-digit code requires major efforts in modifying many application programs. The separation of data and applications would have rendered the change of the two-digit codes a far less expensive effort. The enterprise model extends the 3-schema concept beyond data and applications to provide the separation of the conceptual enterprise model from the physical implementation of technologies, operational and analytic processes.

From the operational side, the enterprise model framework enhances the separation of business processes from the conceptual model. Business functions and the associated data can be defined independent of the organization and process implementation. For example, the processes to implement the business function of “managing customer billing inquiry” can be in the form of a call center operation or in the form of a Web-based self-service process. Yet, the associated business functionality and data requirements for those processes remain the same. Similarly, analytic functions and the associated data can be defined independent of the implementation of analytic processes. For example, the analytic processes to implement the function of “determining the types of products to be promoted to a specific customer” may involve various data mining approaches such as statistical analysis and neural networks, supporting various decision support processes at different levels of an organization. Furthermore, the enterprise model provides the separation of the conceptual data from the technical construct. The conceptual enterprise model provides a unified view of data, irrespective of their physical implementation methods. Therefore, a conceptual data definition of a customer may be physically stored in different places, using different formats. They may include an ERP/Oracle/Unix system in New York, a CRM/DB2/Mainframe system in Chicago and a file system in Hong Kong. The integrity and consistency of data can be achieved through the mappings between the three levels of the Enterprise Model.

Clear Differentiation and Collaboration of Analytic and Operational Functions

It may be intuitively clear that data warehouses are not intended for transactional processing. However, for the lack of understanding or for convenience, many businesses are building data warehouses to satisfy immediate transactional reporting needs. In fact, many IT departments are using data warehousing as an answer to all sorts of data access and reporting requirements. This puts data warehousing in its most ineffective mode of operation. Furthermore, chaos results when every data access and reporting requirement leads to the development of a new data warehouse. The enterprise model framework clearly separates the two domains. The enterprise data model drives the design of the operational data stores that support operational processes and reporting. The analytic data model drives the design of data warehouses that support analytic processes and reporting. Operational processes in the OEM collect critical operational data required by analytic processes in the AEM. The analytic processes create the business intelligence that enhances future operations. The feedback loop allows the learning of the system by validating planned or forecast data with actual data. The system hence acquires knowledge for further analytic processing. As described by Marakas (2003), the components of a decision support system (DSS) consist of the data management system, the model management system, the knowledge engine, user interfaces and users. The enterprise model provides an integrated view of these components of a DSS across the operational and analytic enterprise.

From a Point Solution Focus to an Enterprise Driven Methodology

A point solution focuses on solving a narrow isolated problem. Business intelligence today seldom deals with a narrow isolated problem. More often than not, many aspects of the enterprise data are involved for decision support, making predictions and creating useful business insight. As pointed out by Marakas (2003), a knowledge
base contains information that is domain specific whereas the database and model base components store a wide range of domain-related elements. The process of solving a problem in a narrowly defined domain in decision support may require data and models that span across multiple domains in an enterprise. Lacking the understanding of the needs of the entire enterprise and the omission of an architectural framework constitute the first two of the “seven deadly sins” in building data warehouses described by Kozar (1997). The enterprise model provides the required framework for building data warehouses and DSS.

Consistency of Data across Data Warehouses

The vast number of disparate data warehouses and data marts built over the past years creates problems in data consistency across the enterprise. Many businesses are busy on consolidation projects. They are attempting to consolidate all these disparate data warehouses and data marts into a “common” data warehouse. The unfortunate reality is that as soon as the consolidation project is complete, the new data warehouse becomes another legacy data warehouse that needs to be maintained, synchronized and consolidated to some other data warehouses as the requirements grow. This is equivalent to putting all different files into one big file in file-processing systems. In fact, total physical consolidation of data warehouses is close to being impossible in any practical sense, especially for large organizations. It also defeats the purpose of subject-orientation for building data warehouses. As illustrated in Figure 4, data warehouses and data marts across the enterprise are mapped to their associated ODS(s), which contain consolidated and transformed data from all the sources systems. The conceptual enterprise model provides the integrated framework for the consolidation of data across the enterprise.

Figure 4. Synchronization of Data Warehouses and Data Marts across the Enterprise

Enhancing the Success of New System Development Efforts

Building a new system in an existing enterprise is like building an extension for an existing building. It needs to leverage and integrate the existing infrastructure. For example, a company wants to implement a new CRM package. The CRM package comes with its data schema for “customer” and other data requirements. To implement the package, one needs to build interfaces to the existing ERP and all other legacy, transactional and external systems, each having its own data schema for customers. Each software vendor insists that its customer database is the one that should be used as the “master” database. Complex point-to-point interfaces are required and huge
redundancies are created. That is one of the reasons that most of these implementations are very expensive both in time and costs (typically in the millions of dollars for large companies). The failure rates for these implementations are high. The enterprise model provides the logical framework that leverages the existing information infrastructures. As a result, it reduces development costs and time, while enhancing the quality of the new systems. This applies to building new data warehouses as well as transactional systems. Figure 5 illustrates the mappings from existing systems to new systems through the conceptual enterprise model.

Figure 5. New System Development

![Conceptual Enterprise Model]

**BUSINESS APPLICATIONS**

Two critical criteria dominate the information requirements of today's businesses. The first is the ability to provide real-time access to accurate information across the enterprise to any authorized individual, at any time and any place. The second is the ability to create business intelligence and insight from the mountains of data collected at the transactional level. The first criterion is transactional in nature and the second analytical. The enterprise model addresses both of these requirements in an integrated architecture. While the traditional business performance indicators evolve around internal efficiencies, the new economy requires businesses to extend the value system to their customers, suppliers and partners. Point solutions seldom provide the optimization required at the enterprise level. The following provides two examples that can benefit from the enterprise modeling approach.

**Enhancing Customer Relationship Management**

Two major areas of specific concerns regarding customer relationship management are being addressed in today's businesses: creating customer insight and improving customer services. As pointed out by Todman (2001), the secret to customer relationship management is “to know who our customers are and what it is that they need from us”. Companies are embarking on initiatives to collect and analyze various customer information so that they can better profile and classify customers, predict customer behavior, conduct target marketing, cross and up sell into existing customer base. Information about customers can span across all areas of an enterprise through many different business processes and involving many business units. Information collected through transactions in conjunction with external data can be used for various analytic processes to create customer intelligence that
enhances customer relationship processes. One of the issues many companies have to deal with is to make sense out of the islands of customer data marts to provide such intelligence. Kalakota et al. (2000) described the need for an integrated CRM architecture, where CRM processes are designed around the customer rather than internal functional silos. The portfolio of CRM processes includes cross-selling and up-selling, marketing and fulfillment, customer service and support, field service operations and retention management. Kalakota et al. (2000) further explored the requirements for the integrated CRM architecture requirements, which include the integration of customer content, customer contact information, end-to-end business processes, the extended enterprise and systems. The enterprise model provides the integration framework: the Conceptual Enterprise Model contains the definition of customer content and information, the Operational Enterprise Model contains the design of business processes and organizations, and the Technical Enterprise Model contains the definition of systems. The Enterprise Information Roadmap provides the mappings between these various components. Furthermore, it was pointed out by Preslan (2003) that many CRM projects fail to provide real, reportable business ROI due to the lack of measurements. The inability to align the correct metrics across the business enterprise was a critical reason for such failure. The enterprise model provides an integrated view of customers, enabling various CRM processes where metrics can be defined and measured across the enterprise.

As illustrated in Figure 6, the architecture describing the relationships between the Operational Enterprise Model and the Analytic Enterprise Model can be applied to CRM to create customer intelligence. For instance, the process of a marketing campaign in the OEM collects critical customer data that feeds into the analytic processes in the AEM, which create the customer insight for future marketing campaigns and other CRM processes.

Figure 6. Analytics and Operations in CRM

Enhancing Supply Chain Management

As we move from the production economy to the digital economy, the focus of manufacturing efficiency is shifted to effective supply chain management. Information sharing between various entities in the supply chain is a critical element of successful supply chain management. A notable example as pointed out by Turban et al. (2002) is Wal-Mart’s vendor managed inventory business model. In this example, Wal-Mart shares sales information of each Procter & Gamble (P&G) item in every store with P&G. P&G can then replenish automatically when a certain inventory threshold is reached. Furthermore, such information collected at the point of sales can be used in analytics to create business insight for both retailers and suppliers in many areas including demand and production forecasts, product and facility planning. Wal-Mart’s vendor managed inventory example can be extrapolated to a model of extranets between suppliers and retailers or manufacturers where internal processes and technologies within these business entities are tied together to produce values for the extended enterprise. Furthermore, the lack of accurate and timely information sharing and exchange in a supply chain can cause problems for each member of the extended enterprise. As pointed out by Turban et al. (2002), poor demand information flow may cause the erratic shift in
orders in a supply chain, known as the “bullwhip effect”. This may result in excessive stockpiling and high inventory costs for suppliers. The enterprise model framework provides the integrated roadmap for the design of efficient movement of accurate information between all entities within a supply chain.

**METHODOLOGY**

The methodology in the construction of the enterprise model framework focuses on the virtual integration of the following dimensions: enterprise objectives and goals, data architecture, process architecture, development methods and technical architecture. Traditional data warehousing models while trying to solve domain specific problems often lack the understanding of enterprise objectives and data requirements. Furthermore, traditional data warehousing models are mostly data-centric. The enterprise model framework defines the inter-relationship between the data construct for data warehouses and the analytic processes that enable decision support functions for an enterprise. The enterprise model provides the integration framework for the data management, analytic model management and knowledge management components of decision support for an enterprise, driven by enterprise level objectives and requirements.

In a previous section we discussed the two ends of data warehousing design methodologies using the top-down “big-bang” approach and the bottom-up “loose-confederation” approach. Simon (1998) described a hybrid approach using a distributed data warehouse environment instead of a single monolithic data warehouse used in the big-bang approach. Each data warehouse in the distributed data warehouse layer, sources data from a collection of source applications and feeds a collection of target data marts. As pointed out by Simon (1998), this “component-based” architecture overcomes the deficiency of the two extreme approaches. Most data warehousing design methodologies fall into one of these three categories: the big-bang, the loose-confederation and the component-based. The enterprise model provides the logical framework that optimizes the design of data warehouse components. It enhances the performance and maintenance of the data warehouses and data marts across the enterprise.

The evolution of data warehousing development methodologies follows the trends in software development life cycle. Software development methodologies have evolved from the sequential “waterfall” approach to the iterative “spiral” approach as described by Boehm (1986). Most rapid application development (RAD) methodologies in system development were born out of the iterative concept. Similar to software development, most data warehousing development methodologies adopt either the sequential “waterfall” approach or the iterative “spiral” approach. Various forms of RAD methodologies for data warehousing emerge from the iterative approach. The enterprise model enhances the iterative approach. The independence of the physical and conceptual constructs of the data warehouse allows changes to be made at the conceptual level at minimal cost during the iterative process in the development life cycle. Impacts to the physical design are managed through mappings between the conceptual and physical models. Furthermore, the enterprise model provides the extensibility of the data warehouse design. Various components of the data warehouse can be designed and built incrementally governed by the conceptual roadmap of the enterprise model.

The enterprise model framework emphasizes the interactions between the analytics and operations of an enterprise, while traditional models usually address one of the two aspects in isolation. Furthermore, traditional models often are technology-driven, which put the evaluation and selection of tools and technical platforms before the understanding of enterprise objectives and requirements. Using the enterprise model approach, the conceptual enterprise model drives the technical architectures of data warehouses, which ensures their business usefulness. The conceptual blueprint further protects the design from technology and process changes.

Finally, the proposed framework has been refined and proven based on the author’s many years of consulting practices. Many concepts presented in this paper are refinements of methodologies deployed in actual projects where various aspects of the enterprise model framework have been implemented across different industries. They include organizations in manufacturing, telecommunications, telemarketing, oil and gas, logistics, utility, law enforcement, pharmaceutical, insurance, higher education, state and local government. The enterprise model framework has been used in data warehousing and other system development and integration projects supporting various enterprise initiatives such as enterprise resource planning, customer relationship management and
supply chain management. The benefits of the enterprise modeling approach have been proven in many cases.

CONCLUSION

The digital age has created opportunities and challenges for business enterprises. Rapid changes in market conditions require companies to be adaptive and capable of responding to new requirements quickly. The competitive forces as described by Porter (1980) take on new meaning in the digital age. New entrants and substitution products can be imminent threats. Traditional competitors need to cooperate and share business processes and information. Companies need to extend their value chain to their suppliers, third party service providers and customers. These business conditions may result in major changes in organization structures, business strategies and processes. The challenges are further exacerbated by rapid changes in technologies and incompatible technical infrastructures within the internal and extended enterprise. The enterprise model provides the framework to enhance operational and analytical capabilities for companies to better manage these competitive forces. On the operational side, information access across the entire enterprise based on real-time business requirements becomes a necessity. The enterprise model provides the integrated framework for the real-time management of enterprise information flow. The virtual construct of the enterprise model provides the insulation of the enterprise from process and technical changes and extensions. It provides the necessary framework to afford the economy of scale and scope for an enterprise. On the analytical side, one needs to sift through the mountains of data to find relevant information for decision support, making predictions and creating business intelligence. The enterprise model gives a sound approach to data warehousing providing long term and short term benefits to support analytic requirements for organizations. It provides the cross-domain and cross-functional data construct for decision support. It emphasizes the feedback loop from operations to analytics, enhancing the learning process of analytic processes to produce better business intelligence. In conclusion, the enterprise model serves as the blueprint to create adaptive enterprise-wide information technology architectures to enable business models in the digital age.

REFERENCES


