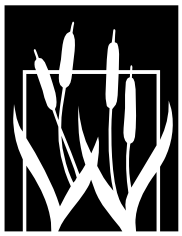

Whitemarsh
Information Systems Corporation

Metadata: The Information Technology Intellectual Property of the Enterprise

Whitemarsh Information Systems Corporation
2008 Althea Lane
Bowie, Maryland 20716
Tele: 301-249-1142
Email: Whitemarsh@wiscorp.com
Web: www.wiscorp.com

Table of Contents

Figures

Tables

1.0 Objective

The overall objective of this paper is to describe the role metadata can play in accomplishing a mission critical enterprise scenario. That is, the transition of a large collection of Human Resources (HR) legacy business information systems and databases to an enterprise-wide HR Enterprise Resource Planning (ERP) environment. This paper further shows how the vast majority of this same metadata can be used to accomplish other enterprise IT scenarios. The paper asserts moreover that the sum total of all this metadata is the enterprise's information technology intellectual property that needs to serve as the foundation for all IT scenarios.

To illustrate these objectives, this paper employs a Legacy HR to ERP migration scenario. The scenario sets out a high level process model, facilitated through identified and collected sets of enterprise-metadata, to interconnect a large collection of HR-focused legacy applications and databases with an already selected enterprise-wide ERP package. The organization in question is world-wide and employs upwards to one million staff in various categories. This scenario is the basis for the material contained in Sections 2 through 6.

Sections 2 explains the nature, scope, and context of the problem at hand, including a description of the essential role that a comprehensive database of metadata plays in this effort.

Section 3 defines the characteristics of the overall environment within which an ERP system must exist.

Section 4 sets out a description of the Knowledge Worker Framework (KWF), why understanding it is critical, and presents a percentage tabulation of the various classes of IT errors that, unless avoided, will cause failure.

Section 5 describes in some detail the three alternatives for accomplishing the effort of deploying the HR ERP package, and then, having settled on the third alternative, presents a high level explanation of that choice. Backing up that choice is hundreds of pages of methodology, books, courses, seminars, and of course, the Metabase System that supports the capture, storage, reporting, and updating all the metadata-based work products.

Section 6 sets out a high level work plan for implementing the alternative chosen in Section 5. This section also shows the critical role played by the Knowledge Worker Framework as it is used to identify the sequence of created work products (columns), their level of detail (rows), and the work activities from the Whitemarsh methodology that cause the work products to be created. The Metabase System and its database is, of course, the database that stores all these work products. All of this both makes this HR ERP Migration effort possible, it also ensures that the complete set of work products are integrated, interoperable, and non redundant.

Accomplishing these work products characteristics is very important because the life span of an ERP is from five to ten times longer than it takes for its initial implementation. If all the work products are not current and continuously maintained, the overall quality of the ERP declines.

Now that the legacy to ERP scenario is complete, Section 7 focuses on generalizing the scenario to show that all the HR-ERP developed metadata is really enterprise-wide metadata that essentially forms IT's intellectual property. Within Section 7, the subsections:

- Describe the processes and models associated with the manufacturing work products and their integration into one overall set of enterprise intellectual property.

- Show how the development of all these work products is accomplished though the varied staff in different organizations.

- Convey a strategy for integrating all the work products into one overall enterprise-level set of intellectual property.

- Identify the different enterprise scenarios that are generally similar in scope and size to enterprise resource planning.

- Set out the advantages and benefits derived from this paper's approach to enterprise information technology intellectual property development and management.

- Indicate a way ahead for similar efforts.

- Show how the information technology project environment needs to change from being project-centric to release-centric.

Section 8 provides examples of where this approach has been accomplished under a number of the different scenarios.

Section 9 presents an overall summary and conclusions.

Section 10 presents a set of references to other Whitemarsh material that can be accessed from the Whitemarsh website as further explanation of the material this paper contains.

Prior to delving into the content if this paper, it must be stated that unless an integrated, interoperable, and non redundant comprehensive metadata environment exists, the probability of success is very close to 0%. Strong words to be sure, but the experiences of many organizations, described in various GAO reports bear this out.

2.0 Essential Infrastructure Environment of Metadata

A key foundation for the process model approach and architecture is the development and deployment of highly organized, integrated, interoperable, and non redundant collection of enterprise-level metadata, which ultimately is information technology's intellectual property, and to use that metadata to:

- Identify and understand the current legacy HR environment.

- Set out an ideal architecture for the target HR environment that is ERP independent.

- Engineer and deploy a commercially viable vendor's HR ERP package based on the collected legacy and enterprise HR model metadata.

- Enable the integration and successful deployment of other enterprise information technology scenarios.

To be truly valuable, these metadata intellectual property collections need to:

- Represent all the critical work products across the effort's.
- Be accessible through commonly available SQL tools and report writers.
- Support the entire effort's community through the Internet.

This paper's definition of metadata as information technology's intellectual property differs significantly from its traditional meaning, that is, data about data. That definition is not only too cute, it is incomplete given the actual word, Metadata. The string, metadata, is divided into meta and data. Meta in the Oxford Dictionary means, "something of a higher or second-order kind." The word, data, however is not employed within this paper in its strictest sense, that is, a data item like Birth date = 03/22/1941, but in more general sense so as to include structured and unstructured data. Unstructured data includes both text and diagrams.

The scope of metadata here is restricted to business information system environments. Not just one, but the entire collection across an enterprise. To restrict a metadata collection to just one business information system is to presume complete encapsulation of mission, organization, function, data and process within well-defined orthogonal metadata collection sets. Simply put, that's not practical, appropriate, nor possible.

Metadata integration and interoperability are essential to a well ordered enterprise. Without integration and interoperability, non redundancy is also not possible. Without all three metadata characteristics, there can only be large tightly bound, parochial stove pipes of enterprise functionality that ultimately are significant reasons for increasing inefficiency, reduced effectiveness, continually lowering profitability, and ultimately, enterprise demise.

Consequently, within the context of this paper, metadata are the materialized artifacts that define the requirements for, the specifications of, design of, and even executing characteristics of the enterprise suites of business information systems, and components within those systems. "System" here is used in a very broad context. Thus, included within the scope of systems are databases, application systems, and their technology environments. Therefore, metadata is all that which is one or more levels of abstraction removed from the actual databases, applications, or their technology environments. In an information technology environment, metadata therefore includes:

- Requirements

- Mission, organizational, and functional descriptions
- Work plans

- Database designs through to schema DDL (data definition language)

- Business information system work products possibly through to computer program source code libraries

- Technology environment designs through to actual installation artifacts

This metadata context, would not include, however:

- Actual databases with data records of employees, invoices, products, and customers
- Executing application systems
- Operating systems and other systems software such as DBMS and Web browsers
- Telecommunications Networks
- Computers

These are not metadata because they are "real," while the previous list represents artifacts "about" these real objects. Once, however, business information systems are executing, metadata may be created "about" the characteristics of the operating business information system environment. That class of metadata would include for example:

- Computer system execution schedules
- Computing resource consumption requirements
- Quantity of records in particular files
- Quantity of users by time of day for particular processes
- Job completion and/or error messages

The metadata management system and supporting metadata database is called the Metabase (i.e., metadata-database) System. An operational example of a Metabase System including its software, demonstration databases, and user guides are available from the Whitemarsh website. While it is not essential that the Metabase System functional data models be previously reviewed to understand this paper, it helps. Thus, it is recommended that the Metabase Overview document be downloaded and the metadata data models be printed and be reviewed to have a solid understanding of the extent of metadata to be captured, interrelated and made non redundant. The link to that is:

http://www.wiscorp.com/metabase/metabase.zip

This paper, does not represent a totally detailed work plan for an enterprise HR ERP migration effort. Rather, it sets out the way-ahead with the full and certain knowledge that the required enterprise metadata design, collection storage, reporting, and evolution software system and database already exist and can be employed on a multi-user Internet-based environment. There are a large collection of references to these materials in Section 10 of this paper..

The enterprise-level metadata environment described in this paper is supported by a thoroughly documented data-centric methodology with courses, seminars, books, white papers, and a set of organizations that have been successful over the past 25+ years.

It needs to be noted that this enterprise-centric metadata approach does not address the following:

- Infrastructure software such as operating systems and database management systems
- Hardware for either clients, servers or networks
- Telecommunication networks or the Internet

These other work products are all addressed by other well known methodologies and are easily integrated with the work products identified and/or referenced in this paper. The approach to build an overall integrated work plan is described in the Whitemarsh Short Paper 12, Manufacturing Project Plans, that is identified in Section 10 of this paper.

3.0 Common Environment

While every environment is unique, this paper presumes the existence of a large collection of application software and supporting databases focused on HR. HR, in the context of this paper, subsumes all employee management processes from staffing requirements through separation and follow-on retirement, all benefits associated with employees including traditional benefits such as compensation, health, life, retirement, education, and the like.

It is presumed that the HR environment has multiple HR software and database systems that are, at best, loosely coupled for the exchange of HR data in support of enterprise-wide reporting and staffing management.

It is further assumed that the HR software may either be through a set of HR applications or a large collection of stand-alone application software systems that either operate through stand-alone databases or an overall integrated database. Given that there are multiple HR software systems and databases, it is presumed that there is not a single set of materials addressing HR architecture, design, implementation, operation, and evolution policies, procedures, and practices.

It is also assumed that there exists multiple HR software and database IT organizations focused on the development, deployment, and evolution of HR software and databases.

This common environment is contrasted to an enterprise HR ERP objective, which is characterized by:

- A single, unified network of HR application software and databases accessible throughout the world.
- A single, unified set of IT development organizations for the development of HR application software systems and databases.
- A single unified set of HR reference data that accommodates the migration of all legacy HR data and proceeds into a unified set of HR data semantics in the future.
- The ability to accommodate both centralized and decentralized HR processes across the world for the enterprise including the ability to support a diversity of human processes and organizations that substantively conform to enterprise-wide HR policies and procedures.
- The ability to create, develop, deploy and maintain supplementary HR functions and data that are integrated with the overall set of enterprise HR functions and data.

4.0 Knowledge Worker Framework: Essential for Understanding

The objective of any enterprise level HR ERP effort is the elimination of large collections of application software and databases. This implies, however, that all the human processes, and HR practices throughout the world will have to conform to one unified set of HR application software and procedures. Attempts like this (taken from http://www.erp.asia/erp-failures.asp) have almost always failed because:

- Failed cross-representation agreement on enterprise wide business processes.
- Lack of visible, vocal and meaningful executive sponsorship.
- Lack of formal and disciplined project management.
- Project team turn-over of key staff.
- Inability to identify and mitigate risks or remedy incidents which ultimately escalate.
- Insufficient training.
- Troubled user adoption.
- Too much software customization.
- Project viewed as an “IT” project.
- Dirty data

Supporting these failure causes are those cataloged from an analysis of U.S. Government Accountability Office (GAO) studies of failed large-scale multi-$100 million IT efforts. Table 1 sets out a framework highly tuned for the knowledge worker. The rows were directly taken from the Zachman Framework for Information Systems. The columns are different in order to make them relevant to enterprise-level efforts and to maximize success probability.

The contents of each cell are various work products. These work products are created by members of project teams. The key to make these work products valuable is that they are integrated, interoperable, and non-redundant collections of metadata stored an enterprise-wide database for the knowledge worker.

The errors cited by the GAO studies were counted and turned into percent values. These percents, set within the Knowledge Worker Framework's rows and columns, are set out in Table 2.

The bottom line message for all these percent values is that if the work products identified in a cell are accomplished and if these work products are created such that they are integrated, interoperable and non redundant with all the other work products, the cell's percent of failure is eliminated. So, if the first row is accomplished acceptably, there is an 18% reduced chance of IT system failure.

The implications from Table 2 are summarized and presented in Table 3. The message is clear and compelling: create the work products, cause them to be integrated, interoperable, and non-redundant or IT system failure is sure to occur.

This mirrors John Zachman's famous saying that he's stated to hundreds of conferences since the late 1980s. That is:

> *Someday you are going to wish you had all those models, Enterprise-wide, horizontally and vertically integrated at an excruciating level of detail.*

That "someday" probably occurred about 25 years ago when it first became possible to have a unified database of infrastructure work products. And that "someday you are going to wish" also probably occurred about 25 years ago when it became possible to have enterprise-wide IT systems and databases. Enterprises that acted on Zachman's admonition often succeeded at their IT efforts, and those that did not almost always failed. The United States Government's General Accounting Office's library is a graveyard of failure tombstones (a.k.a., "GAO Reports").

Knowledge Worker Framework						
Perspective	**Mission**	**Database Object**	**Business Information System**	**Business Event**	**Business Function**	**Organization**
Scope	Business missions	Major business resources	Business information Systems	Interface events	Major business scenarios	Organizations
Business	Mission hierarchies	Database Domains, and Resource Life Cycles	Information sequencing, hierarchies, and use cases	Event sequencing and hierarchies	Business scenario sequencing and hierarchies	Organization charts, jobs and descriptions
System	Policy hierarchies	Data Elements, Specified Data Models and Identified Database Objects	Information system designs	Invocation protocols, input and output data, and messages	Best practices, quality measures and accomplishment assessments	Job roles, responsibil-ities, and activity schedules
Technology	Policy execution enforcement	Implemented Data Models and Detailed Database Objects	Information systems application designs	Presentation layer information system instigators	Activity sequences to accomplish business scenarios	Procedure manuals, task lists, quality measures and assessments
Deployment	Installed business policy and procedures	Operational Data Models	Implemented information systems	Client & server windows and/or batch execution mechanisms	Office policies and procedures, to accomplish activities	Daily schedules, shift and personnel assignments
Operations	Operating business	View data models	Operating information systems	Start, stop, and messages	Detailed procedure based instructions	Daily activity executions, and assessments

Table 1. Knowledge Worker Framework.

Knowledge Worker Framework							
		Man-Machine Interface					
		Machine		Interface	Man		
Deliver-ables	Mission	Database Object	Business Infor-mation System	Business Event	Business Function	Organ-ization	Row Pct
Scope	5	2	3	1	3	4	18
Business	5	3	2	1	6	6	23
System	3	2	2	1	12	8	28
Technology	1	0	0	0	8	6	15
Deployment	0	0	0	0	5	5	10
Operations	0	0	0	0	3	3	6
Col. Pct	14	7	7	3	37	32	100

Note: All numbers expressed as Percent allocations of errors to cells ...12 Gray cells are IT Cells

Table 2. GAO IT System Errors expressed as percent values

Projects/Scenario Focus	Knowledge Worker Framework Rows and Columns	Percent of Errors Avoided
Building and Employing Enterprise Architecture Models	*Scope and Business Rows*, all columns	41% of all GAO IT errors.
Creating and Evolving Information Systems Plans	*System Row*, Mission through Business Information System columns.	About 8% of GAO IT errors
Architecture and Engineering of Data Models	*Technology Row*, database object column	Less than 2% of Errors
Performing Reverse Engineering of Legacy Systems and Databases	*Technology Row*, database object column.	Less than 2% of Errors
Forward Engineering Manufacture of New Systems and Databases	Operations, Deployment, Technology and System Rows. Database Objects Column.	Less than 2% of Errors
Employment Errors	Systems through Operations Rows. Operations and Functions Columns.	50% of Errors.

Table 3. Percent Summarization and Source of IT Errors.

Figure 1 shows the overall interrelationship among the Knowledge Worker Framework work products accomplished across all its six columns and down its six rows. The way to understand the diagram is to first understand that the knowledge worker columns (Mission, ..., Business Organization Model) are all related to each other through many-to-many relationships. The basis of the relationship is the "box" that is directly above the Knowledge Worker Framework column.

The columns and their interrelationships are:

- **Missions**. Missions are representations of the idealized hierarchical descriptions of enterprise outcomes. Missions are devoid of both "who" and 'how." Hence, apolitical. It is upon missions, that database objects (that includes all data specifications) are based. Because database objects and business information systems are founded on missions, they are longer lasting, more comprehensive, apolitical, and are better able to be integrated and non redundant, and evolved.

- **Database Objects**. Database objects represent the persistent memory necessary to fulfill one or more enterprise missions. Similarly, a mission can be fulfilled by one or more database objects. These are defined within the context of idealized enterprise missions that are long lasting, apolitical, and independent of the ever-changing business functions and organizations. These are defined in the context of database objects. Database objects are defined within the context of missions, they too are long lasting and apolitical because they are also independent of ever-changing business functions and organizations.

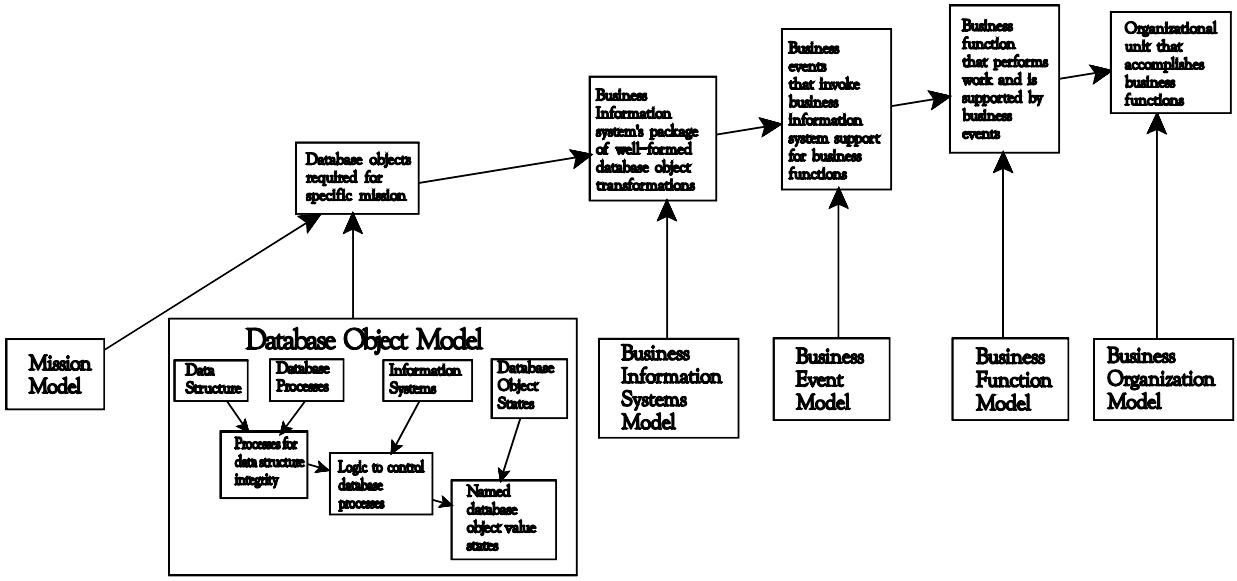

Figure 1. Interrelationships among Knowledge Worker Framework columns and rows.

- **Business Information Systems**. Business information systems are the application information systems necessary to collect, store, report, and update the data necessary for the database objects. These are defined within the context of like database objects and so, for the very same reasons, these are long lasting and apolitical.

- **Business Events**. Business event models are the interface events that enable one or more business functions to employ business information systems. In turn, a business event may be fulfilled through the employment of multiple business information systems. The business event models are tactically and operationally based because they are designed to reflect the changing needs of business functions. Multiple business functions that are essentially equivalent are able to be mapped to the same business event, which, in turn are mapped to the same business information systems.

- **Business Functions**. Business functions, which are human processes, are accomplished by business organizations in the use of business information systems to collect et al the data necessary to accomplish enterprise missions. The business functions are both tactical and operational and also ever changing and political. Needless to say, organizations too are both tactical and operational and are thus ever changing and political.

- **Business Organizations**. Business organizations are bureaucratically engineered collections of individuals within enterprises that have a common functional purpose. While business organizations often have highly engineered functions, there are functions that are common across business organizations. Because of commonality, the relationship between business organizations and business functions is many-to-many. This enables common definition, integration, and non redundancy. It also enables common use of Business Events, Business Information Systems, and Database Objects, and Missions.

Because the first three columns (down through their rows) work products are long-lasting and a political and the last two columns (business function and business organization) are tactical and operational and are thus ever changing and political, the first three columns are accomplished first. The last two columns are accomplished second, and the business event column is accomplished last and serve as the intersection mechanism between the apolitical columns and the political columns. Because of this architecture, the end-result is longer lasting, more comprehensive, apolitical, and are better able to be integrated and non redundant, and evolved. Win-win for everybody.

5.0 Alternatives for HR-ERP Migration

This paper sets out three overall alternatives for accomplishing ERP migration from stand-alone, stove-pipe HR applications and databases to a unified enterprise-wide ERP-based HR

environment. Each of these alternatives is briefly presented with a focus on their affect on human processes. Alternative 3 is chosen and at a high-level, is detailed over 10 pages.

Central to this approach is the Knowledge Worker Framework. That framework's identification and summarization of failure percents set out in Table 2, and this paper shows that 90% plus of the failures occur within the Scope and Business rows, and the activities from the Business Function and Business Organization columns that occur after a business information systems has been developed and deployed. Consequently, the real focus of any ERP effort must be on these work products. The three alternatives for accomplishing the ERP migration work are:

- Replace all the legacy systems with the deployed HR-ERP system
- Directly connect the legacy systems with deployed HR-ERP system
- Indirectly connect the legacy systems through an Enterprise HR Model to HR ERP

5.1 Replace Legacy Systems with HR-ERP

This alternative has the immediate attraction of eliminating large collections of application software and databases. This implies, however, that all the human processes and HR practices throughout the world will have to conform to one unified set of HR application software and procedures. Attempts like this (taken from http://www.erp.asia/erp-failures.asp) have almost always failed because:

- Failed cross-representation agreement on enterprise wide business processes.
- Lack of visible, vocal and meaningful executive sponsorship.
- Lack of formal and disciplined project management.
- Project team turn-over of key staff.
- Inability to identify and mitigate risks or remedy incidents which ultimately escalate.
- Insufficient training.
- Troubled user adoption.
- Too much software customization.
- Project viewed as an "IT" project.
- Dirty data

The only time this alternative has been successful was with a very restricted and geographically homogenous, highly top-down, command-and-control environment wherein the only valid objection entertained for not "doing it the ERP way" was that the end-user would go to jail. The project manager also decreed that there would zero changes to the ERP system unless and until it could be proven that not "doing it the ERP way" would cause someone to be arrested and sent to jail.

If this ERP migration alternative is taken, all the policies, procedures, processes and legacy systems and their databases that had been designed, developed and implemented from the

early 1970s onward would have to be gathered, analyzed, and entirely redeveloped to conform to those contained in the ERP system.

For these reasons, and based on the many experiences from both from industry and the GAO IT failure audits, attempting to re-cast all the existing HR policies, procedures, and best practices across all the myriad of geographic regions, organizations, and stylized implementations of legacy HR will lead almost inevitably to failure. Thus, alternative 1 is judged to be not possible.

A final reason why this alternative is unacceptable is that the very character of HR for the enterprise has to be transformed to be that of the ERP vendor. Such a drastic change in the character of the enterprise has a very high likelihood of failure. And further, if the enterprise ever decides to change from on ERP system to another, all these culture wrenching activities have to be accomplished all over again.

5.2 Direct Connect Legacy Systems with HR-ERP

This alternative, Direct Connect Legacy Systems with HR-ERP, is slightly better than the first. It requires the transformation of all existing legacy systems and databases to conform to the processes and database structures of the ERP system. This could be a very large effort and would ultimately result in the creation of a very large quantity of interface systems and intermediate databases that would have to be created and maintained between the legacy systems and their databases and the HR ERP system's database and application system environment.

The ultimate result again is that the legacy systems and databases would become captive to the ERP vendor's systems and databases. This process would have to be repeated if for some reason the ERP HR vendor needed to be changed. Consequently for these reasons this second alternative is judged unacceptable.

5.3 Indirectly Connect Legacy Systems through Enterprise HR Model to HR ERP

The final alternative, indirectly connecting legacy systems through enterprise database to HR ERP is judged acceptable for these reasons:

- There is an enterprise-centric HR model that is the union of all the enterprise's best practices in terms of HR processes and databases. This model would retain the characteristics of the enterprise that would have been created over many years. There is value in the "tried and true."

- There is one set of data transformations between the legacy systems and those required by the enterprise HR models. This one set of transformations will be fewer in number,

better able to be efficiently and effectively specified, and over time more efficient to maintain.

- There is one set of data transformations between the Enterprise HR model and the HR ERP vendor's processes and databases. For the same reasons cited above, the quantity of transformations will be fewer, each transformation will be simpler, and in the event that there is ever a change in ERP systems, there will only be one set of transformations needed between the enterprise HR model and that of the replacement ERP HR system.

The architecture of this alternative is depicted in Figure 2.

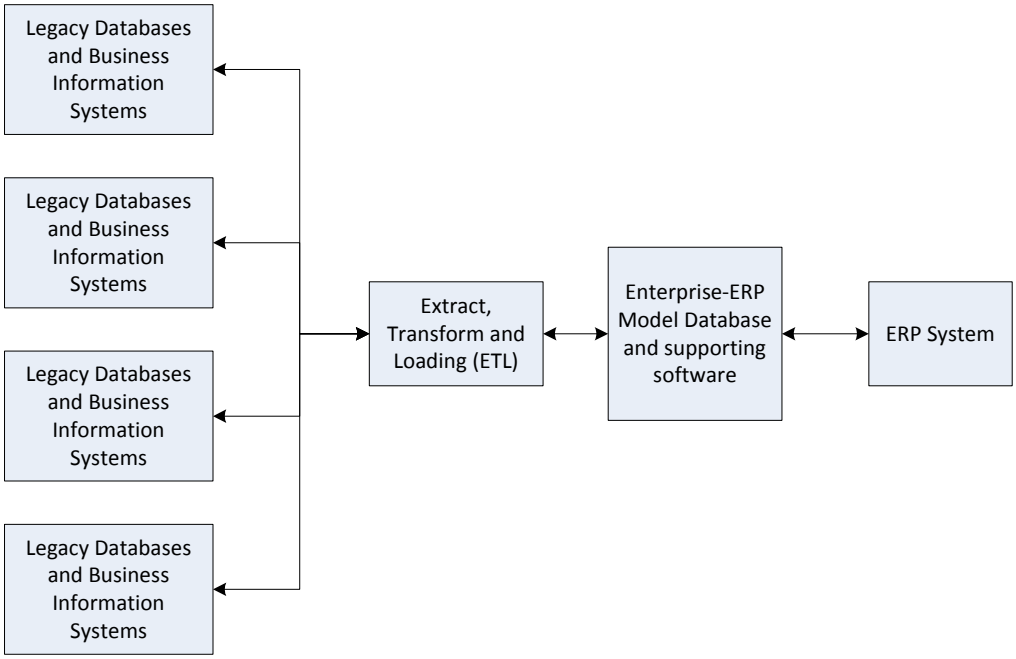

Figure 2. Architecture diagram of Alternative 3, Indirect Connect Legacy Systems through Enterprise HR Model to HR ERP

Achieving Alternative 3 is not easy, however. That is mainly due to the list of failure opportunities listed in Section 5.1 above. While the entire list is essential to deal with, this paper focuses on the following four:

- Failed cross-representation agreement on enterprise wide business processes.
- Troubled user adoption.
- Too much software customization.
- Dirty data.

5.3.1 Detailed Cross-Representation Agreement on Enterprise-wide Business Processes

A short story to address this topic is appropriate here. In the early 1990s, during a project that had as its key products: 1) Enterprise architecture, 2) Enterprise data models and 3) Enterprise information systems plans, a key activity was to develop the "missions" across the enterprise. See Column 1 of Table 1, Mission, of the Knowledge Worker Framework for an idea of the scope of the Missions effort.

During one set of the interviews in Bedford, Massachusetts, the descriptions of the mission areas for Finance was detailed, reviewed, and completed. It is important to this "story" to detail the key distinction between Mission and Functions presented in Table 4.

Key Differences	
Mission	**Function**
Mission descriptions are idealized descriptions of the characteristics of the end result. Mission descriptions are noun-based sentences.	Functions are descriptions of how to accomplish an end result. Functional descriptions are verb-based sentences.
Mission descriptions are a-political. They are devoid of who and how. There should only be ONE mission description for a mission.	Function hierarchies are commonly tainted by organizations and styles. There could be any number of equivalent functional statements for a given function.
Databases and Business Information Systems are based on missions.	"Human" activities and organizations are based on business functions. You "BPR" (Business-Process-Re-engineering) functions
Mission descriptions are strategic and long range.	Functions are tactical to operational, medium to short range, and are organizationally sensitive.

Table 4. Key differences between Missions and Functions.

Upon returning to the Washington, D.C. area, the Washington's office of Finance wanted a report on the results accomplished in Bedford. The mission statements were presented. The Washington folks went "nuts." They said, "how dare they want us to do things the way those 'Neanderthals' in Bedford do."

A quick review uncovered what had not been done. It was to recast the Bedford Finance Mission Descriptions to be:

- Cast in terms of "Idealized" outcomes.
- Removal of all organizational, that is, the "Who" references.
- Removal of all actual process, that is, the "How" descriptions.

Quickly, the mission statements were adjusted and the results re-presented. There was unanimous acceptance of the results. We "finally" got it! These recast results were also sent to Bedford for review and acceptance. Their verdict was "That's much better, good to know that you Washington folks can do something right."

Key are the fundamental questions that the columns of the Knowledge Worker Framework answers and the relationships among the columns. Knowledge of this and accomplishing the work in a specific order eliminates many if not most of reasons for failure. Table 5 provides these questions and interrelationships.

Knowledge Worker Framework Key Questions and Interrelationships	
Column	**Questions and Interrelationships**
Mission	What are the essential missions that define the very existence of the enterprise, and that are the ultimate goals and objectives that measure enterprise accomplishment from within different business functions and organizations?
Database Object	What data is needed by functional proponents, how is it defined within data architectures and databases and how and where are those databases deployed and then used by business information systems in support of mission accomplishment?
Business Information System	Exactly what are the business information systems, where are they, how are they related to mission, organization, function, and databases. What is the impact on these business information systems when policy (a.k.a., data) is required or changed?
Business Event	What are the key business events and collections of events set within business and calendar cycles that enable business functions to interact with the business information systems that collect, report, and update the data essential to enterprise missions?
Business Function	What procedures are performed by groups in their achievement the various missions of the enterprise from within different enterprise organizations?
Business Organization	Which organizations are accomplishing what aspects of missions with what databases, information systems and through which functions?

Table 5. Knowledge Worker Framework Key Questions and Relationships.

It is a longstanding tradition that the traditional life cycle for business information systems development proceeds from:

- Requirements, which then encompasses ...
- Business Functions, which then encompasses ...
- Business Information Systems, which then encompasses ...
- Database Designs

This four-part approach has the following problems.

- Requirements can never be fully know. They unfold, are discovered across time, and change. Hence they can never be truly fixed. If they are declared "final" before fully known, the only thing certain is failure. What is true, however, is that requirements changes cause cascading cycles of business function, business information systems, and database design changes.

- Business functions are inherently political. They are process/activity based, are constrained by the actors, environment, organizational structures, and available resources. If any of these change, so too do the business functions change. As with requirements changes, business function changes cause cascading cycles of business information systems, and database design changes.

- Business information systems are almost never understood completely before they are estimated, sized, or started. Thus, they are almost always late, are impacted by changes in personnel, requirements, budgets, and other available resources. Similarly, changes in business information systems cause changes in database design changes.

- Database designs, especially given the preceding three encompassing and changing: requirements, business functions, and business information systems, are seldom elegant, well designed, or are configured to be independent from the every changing requirements, business functions, and business information systems.

How then does such an environment square with a database's fundamental definition: ***A treaty that governs the behavior to its users?*** Such a definition presumes that a database's design should be much higher in the four-part traditional pecking order.

Clearly a database's design must be set against "requirements." But how can that be when "requirements" unfold, are discovered across time, and change? The answer is simple: Base the database design on Mission. Well done, missions are apolitical and very long lasting. Missions are devoid of who and how. Missions are the idealized, hierarchically organized textual descriptions of the essence of the enterprise. They are inherently apolitical.

This approach, which has been proven over and over through the years as efficient, effective, and accurate is as follows:

- Missions, which then encompasses
- Database domains, which then set the stage for...
- Database designs, that enable the high level specification of...
- Business information systems.
- In parallel with Missions, the Business Organizations, which enable the discovery of...
- Business Functions, which, together with Business Information Systems, are...

- Intersected through Business Events.

These work products fill out the Scope and Business rows of the KWF. To ensure accuracy, and to ensure that all relevant "requirements" are teased out into the daylight, robust prototyping through a high quality business information system generator like Clarion from SoftVelocity (www.SoftVelocity.com) is employed. With a business information system generator like Clarion, what traditionally takes from months to years to be developed can be accomplished in days to a few weeks or to at most a month.

So, the first "order of business" is Missions, which is the left-most column in the Knowledge Worker Framework. Next are the database designs that are fitted within the scope of missions. These too are idealized to meet the needs of enterprise missions, are independent of business organizations and functions. Thus, they too are apolitical.

Thereafter come business information systems that are created to fulfill the needs of the database designs. It must be noted, however that the only business information systems that must be created, at this time, are those need by the business functions through their intersecting business events. Whole swaths of a database's design can stand waiting until some future cycle of needed business information systems. To do this is just plain, good architecture. Better to have an expandable house on the drawing board than to continuously tear what's there down and start over, and over, and over.

Proceeding from the right-to-left direction from the Knowledge Worker Framework are Business Organizations and Business Functions. The "OK Corral" is Business Event. That's where the 'apolitical' missions, database designs, and business information systems meet the business functions that are set within organizations.

Not only is this strategy inherently more logical, coherent, and long lasting, it has never failed over the past 30 years.

An additional side benefit from this approach is that it enables John Zachman's admonition to be realized. That is, *Someday you are going to wish you had all those models, Enterprise-wide, horizontally and vertically integrated at an excruciating level of detail.* With this approach, that admonition can be accomplished efficiently, effectively, and set within a context that it can govern future work. Figure 2 depicts a very high level metadata and interrelationship architecture. The actual metadata data models that embrace the totality of this effort are presented in the Metabase Overview document. The link to that document is:

http://www.wiscorp.com/metabase/metabase.zip

During the early stages of ERP architecture efforts, the various organizations and functions need to be identified, interviewed, work products created and analyzed. These work products are stored in the Metabase System's database. As the efforts progress, the individual aspects of the work products that are essentially the same can be interrelated through the network-structure

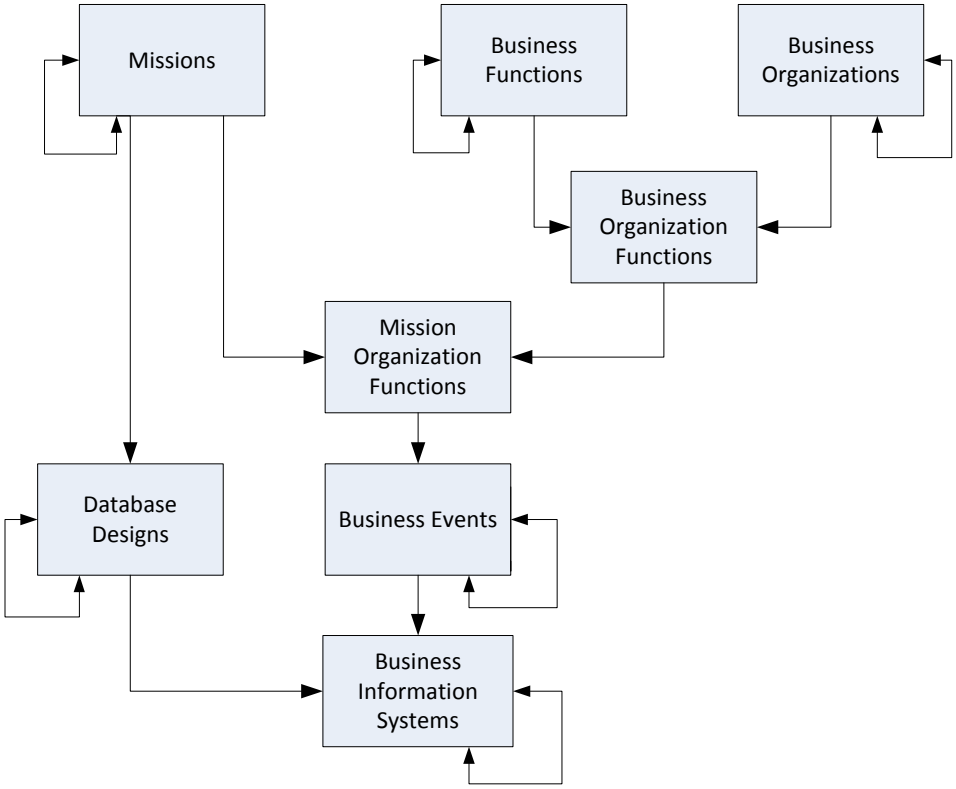

Figure 3. High level diagram of metadata driven approach to effective ERP deployment

capabilities of the metabase database and software. The network relationships across peer-level sets of work products are depicted through the many-to-many relationships associated with the key metadata objects in Figure 3.

Over time, the work products across all the columns from the first two rows of the Knowledge Worker Framework, that is, Scope and Business are captured, reviewed, and stored, in an integrated, interrelated and non redundant fashion. It is neither possible nor essential that there be one and only one business organization and/or business function model. To force such a requirement is to guarantee failure. This is unnecessary because these various models are interrelated across the Knowledge Worker Framework's columns, and rows, and even within these columns and rows in a many-to-many fashion.

Related to the Bedford and Washington example above, the approach did not require that either organization change their organizational structure or their methods (i.e., functions) through which they conducted business. Since neither the database designs nor the business information systems architectures were encapsulated within the business functions, that, in turn were encapsulated withing business organizations, a level of independence was able to be preserved. The Business Event column represents the ultimate intersection of the apolitical

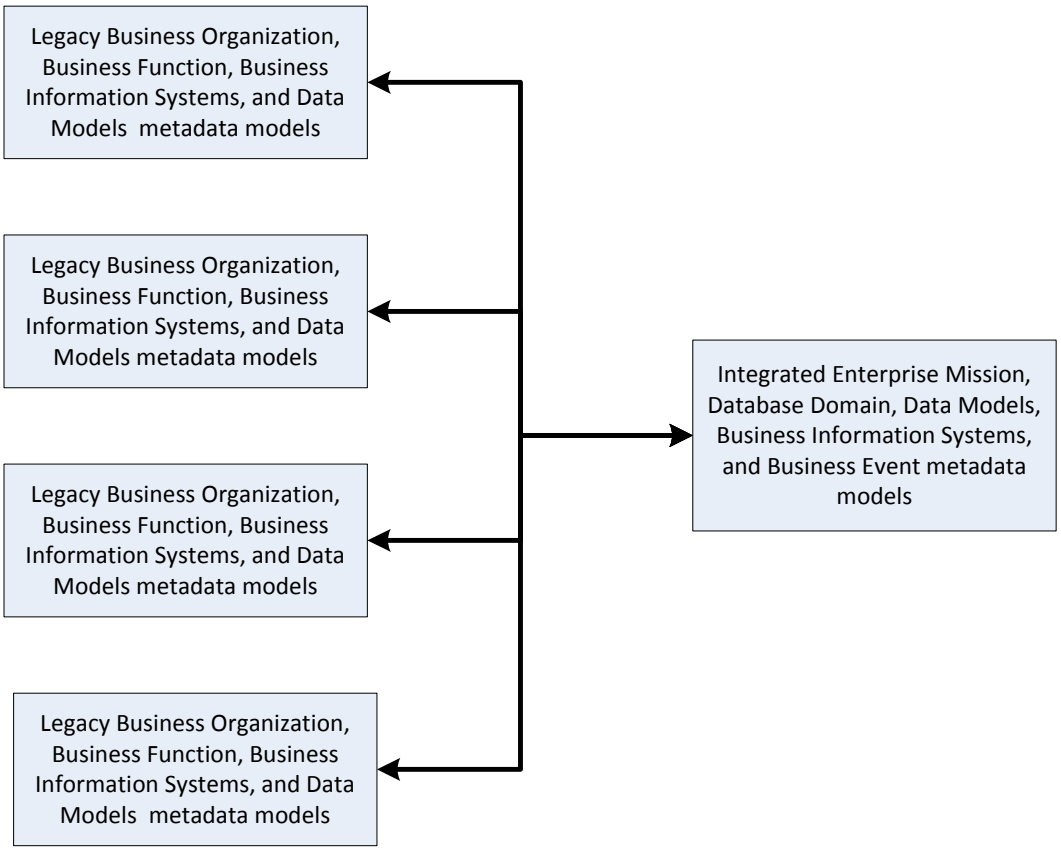

Figure 4. Integration of metadata identified, captured from legacy and inductively determined enterprise ERP HR model.

mission, database designs, and business information systems with the political business organizations and business functions.

The result of this approach is that the work products for organizations, functions, information systems and database models are stored in the metabase and are interrelated are depicted in Figure 4. The uncovered legacy models are on the left, and the determined enterprise models are on the right. During this effort, legacy models are captured, and subsequently, they are mapped to the evolving enterprise model.

As each legacy HR organization is identified and their HR artifacts are identified and analyzed, these uncovered artifacts are entered into the Metabase System in the metadata models including:

- Business Information Systems and Business Events
- Data Elements
- Database Objects

- Document and Form
- Information Needs Analysis
- Mission, Organization and Function
- Requirements
- Resource Life Cycle Analysis
- Specified, Implemented, Operational, and View Data Models
- Use Case

Note that in this approach, the various identified legacy models are not replaced. Rather, they are accommodated and interrelated with the evolving enterprise HR model. The legacy work products are included for the following reasons:

- Likely, significant efforts have been expended in the development of the organizational and function models. These efforts need to be recognized and preserved if at all possible as a deference to local traditions, policies and procedures. Remember, it is due to ignoring customs, mores, and traditions that cause 50% of all the IT systems failures. Tragically, these failures occur after replacement systems are deployed. To spend 100% and get nothing is certainly not a good strategy.

- Likely, significant efforts have been expended in the development of data models. These data models are a source for names, data types, lengths, data-centric business rules, and most critically, value domains. These need to be, at the very least, accounted for in any enterprise data model. One past example of not paying appropriate attention to value domains resulted in nine gender codes. 9? Yes, 9!

- Likely, significant efforts have been expended in the determination of various business and calender business event cycles for data capture, summarization, and reporting. If these are not accommodated in any final solution, while there might be “truth” in individual data values, management reports will likely appear different and incorrect.

5.3.2 Troubled User Adoption

Because of the approach presented in this alternative, there is active end-user participation during the process of legacy discovery, understanding, and incorporation of metadata within the Metabase, the probability of having troubled user adoption is greatly lower. That is because of the direct involvement by the users. Not only are they involved, they are able to see, review, and comment on their legacy models, the mapping to the evolving enterprise HR models and the actual “living and breathing” prototypes of the new HR functionality. This will go to great lengths to significantly reduce the after deployment end-user rejection.

5.3.3 Too Much Customization

There is likely to be little if any customization because of the "indirectness" of mapping a determined Enterprise HR model that serves as a comprehensive set of operating and already validated requirements through prototyping. The totality of the functionality will be represented in the enterprise HR models, which, in turn, are mapped to the functionality of the ERP HR models.

5.3.4 Dirty Data

This last item can be a real killer. In the deployment of one ERP effort, there were nine (9) genders. In another effort, there were five, six, and seven "digit" Social Security Numbers. Some Social Security Numbers are just string with "hyphens" while others had no "hyphens" and were a numeric data type.

An advantage of having a numeric data type on Social Security Numbers is that you can add and subtract them, determine their mean, mode, average and standard deviation. (That was hopefully seen as a joke, of course.)

The list goes on and on. Wrong addressed, birth dates, zip codes, a multiplicity of mistaken but close names, persons with different addresses and the same phone numbers, three pages of different configurations of "Yes" and "No" indicator codes, and of course, allegedly dead persons who are not only still on the payroll but are walking the halls.

One organization within the same enterprise had the ability to choose 35 different employment separation codes while another organization had the choice of just two: not separated, separated.

What to do?

While the answer is quite simple, that is:

- Identify the value domains,
- Capture and store them into the metabase,
- Resolve differences, and
- Deploy solution

The process, however, is long, tedious, and likely the cause of much friction. There is a need to capture the existing value domains which can be in multiple places including DBMS schema value domain clauses, whole table structures, program source code, and through end-user memory and documentation.

Once uncovered, the meanings of each of the restricted values need to be set down. Are there overlaps? How to resolve them? If there are multiple code sets across the enterprise for the same business fact, the differences must be identified and resolved.

At the very least, the strategy contained in this paper is able to be accomplished and the various value domains are able to be uncovered, stored, reviewed, and mapped one value set to another. The paper, Reference Data Management, accessible through a citation in Section 16 details an approach to address reference data.

6.0 Key Components and HR ERP Migration Phases

Three alternatives for implementing the HR ERP were analyzed in Section 5. Alternative 3, Indirectly Connect Legacy Systems through Enterprise HR Model to HR ERP, was chosen. This alternative requires the identification, analysis, understanding and specifications of three environments, which are:

- Legacy HR databases and information systems.
- Enterprise HR Model.
- ERP HR Database and information systems.

Once these three environments are understood and recorded in the Metabase System database, the intersection of Legacy and Enterprise HR Models, and the intersection of the Enterprise HR Model and ERP HR Models can be accomplished.

Collectively, all the work products accomplished through the Whitemarsh methodology are identified within with the Knowledge Worker Framework. As work products are generated they are recorded into the Metabase System's database in an integrated, interoperable and non redundant fashion. To accomplish the HR ERP Migration the five work phases are:

- Build Legacy HR Models.
- Build Enterprise HR Model.
- Build ERP HR Model.
- Intersect Legacy HR with Enterprise HR Model.
- Intersect Enterprise HR Model with ERP HR.

Each work phase produces its own collection of Knowledge Worker Framework work products and sets up the accomplishment of the next phase. Once accomplished and stored in the Metabase System database, the work products are intersected on many levels so that required transformations in processes, semantics and data can occur.

Figure 5 presents the overall work flow among the first three major phases. The constructed metadata-stored models are shown as circles and the Metabase System and Database are at the center acting as the storage, integrating, interoperable, and non redundancy enabler.

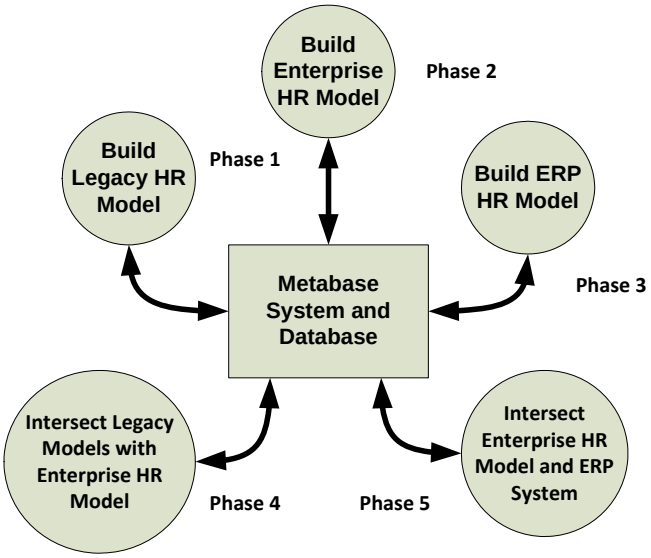

Figure 5. Major HR ERP Migration Phases

The two intersection phases are accomplished through the Metabase System and are represented by the bi-directional arrows not only within the Metabase System icon itself but also between the Metabase System and the three phases. The two intersection phases also result in the creation of the databases and business information systems appropriate to set the HR ERP environment into production.

The Metabase System's technical processes used to build these work products are described in the various Metabase System user guides. The data models for the Metabase System database is presented in ER fashion within the Metabase Overview document. The data model diagrams within the Metabase Overview document are all in a "third normal form" design. That is, each rectangle is a database table.

6.1 Overarching Project Methodology

The methodological basis for accomplishing the work products across the five HR ERP Migration phases is presented in the Whitemarsh methodology's work breakdown structure. Essentially, the Whitemarsh methodology is accomplished multiple times, once for each Legacy HR model, once for the Enterprise HR model, once for the the ERP HR model, and then once for each intersection, that is between Legacy HR models and the Enterprise HR database, and once from the Enterprise HR database and the ERP HR system. During each execution, the work products are stored in the Metabase Systems database in an integrated, interoperable and non redundant fashion.

The Whitemarsh methodology is divided into six phases, which are:

- Preliminary analysis.
- Conceptual specification.
- Binding.
- Implementation.
- Conversion and Deployment.
- Production and Administration.

Table 6 presents brief descriptions for each of the Whitemarsh methodology phases. The work breakdown structure for the Whitemarsh project methodology is completely detailed in a myriad of materials on the Whitemarsh website.

Whitemarsh Database Project Methodology	
Phase	**Brief Description**
Preliminary Analysis	The Preliminary Analysis Phase scopes the effort, creates the mission model, a high level data model, a business function model, and after functional experts review, determines if the proposed project effort is practically and politically possible.
Conceptual Specification	The Conceptual Specification Phase starts with the go-ahead for the effort and produces a complete, validated through prototyping, implementable specification of the entire project effort.
Binding	The Binding Phase takes an accepted conceptual specification and produces a database project design and other critical project components that are married to the selected DBMS environment.
Implementation	The Implementation Phase accomplishes through DBMS, code generators, report writers, and ancillary product development (training, hotline, and the like). A complete, ready for production, business information systems and databases. The product of the fourth phase is a thoroughly tested system, ready for production.
Conversion and Deployment	The Conversion and Deployment phase takes the production-ready business information systems and databases, installs them and performs all the appropriate data conversion efforts and tests them until fully ready for production. It also includes hardware and software procurement and installation, and the accomplishment of all production use end-user training.
Production and Administration	The Production and Operation phase begins by putting the business information systems into production. During production operations, emergency and standard maintenance, and information systems evolution is accomplished. Finally accomplished is a thorough audit in support of lessons learned.

Table 6. Whitemarsh Project Methodology Phases and Descriptions.

Preliminary Analysis. The overall objective of the Preliminary Analysis phase is to determine whether the project is politically and practically possible. In addition, the work products enable an accurate estimate about the remaining phases. The work products from this first phase are able to be quickly accomplished and presented to functional experts to ensure that work in the subsequent phases are well founded.

Once the work of the Preliminary Analysis phase is completed it can be stopped. Even if stopped, however, the work products contribute significantly to the development of enterprise architectures. Given the existence of the Metabase System database, and the existence of earlier work efforts, new efforts can be started that can take advantage of these completed work products. These work products and the Metabase System enable knowledge work to be accelerated over time.

Conceptual Specification. The overall objective of the Conceptual Specification phase is to detail the work products contained in the Preliminary Analysis phase. Completely specified are the necessary details for the implemented data models, data update, data conversion, and the necessary architectures to support reporting. The specified level of detail is sufficient to pick the appropriate DBMS. Detailed as well are the requirements for the various System Control functions, which include audit trails, logical and physical data reorganization, backup and recovery, and the like.

Similar to the Preliminary Analysis phase, work can be stopped after the Conceptual Specification phase is completed. The phases's work products, stored in the Metabase System can be of great value to other functionally interdependent efforts.

The next phase, Binding, is one of indeterminate length mainly because of the need to select, install, test a DBMS, and integrate its characteristics into the appropriate Conceptual Specification work products.

Binding Phase. A key objective of the Binding Phase, given the completed work products from the Conceptual Specification phase is to pick a DBMS that supports the requirements and design artifacts expressed during the Conceptual Specification phase.

Once the DBMS is picked, the DBMS-affected specifications can be finalized. These include physical database designs for indexes, buffers and the like, data loading and updating, and most of the areas associated with System Control. That is: backup and recovery, audit trails, security and privacy, message processing, logical and physical reorganization, application optimization, installation and maintenance, multiple database processing, and concurrent operations.

Since the selection, procurement, and testing of a selected DBMS can take an extensive period of time, the HR ERP migration effort may come to a standstill until the DBMS is available for assessment and thereafter for project implementation.

Implementation Phase. The Implementation Phase mirrors a traditional data-centric implementation effort. In addition to planning, the major sub-phases are Logical, Physical, Interrogation, and System Control.

During the Logical sub-phase, the database designs are completed and submitted to the DBMS for compilation. Included are all aspects of the database's designs including adjustment of the upper-layers of data specification such as database domains, semantics, reference data, value domain specifications, and view specifications.

The key components of the Physical sub-phase are the database's storage structure, indexes, relationships and data storage architectures. Included as well are data update and conversion subsystems. In the case of the HR ERP Migration, there is an extensive ETL (extract-transform-load) efforts between the legacy business information system and databases and the Enterprise HR database, and then between the Enterprise HR database and the chosen ERP databases.

The Interrogation sub-phase includes data report and even data warehouse specifications. In the case of data warehouses, additional data update subsystems need to be identified, designed and implemented to ensure that management reports are consistent across a fixed period of time. Careful attention must be paid to time, precision, and granularity specification.

The System Control sub-phase implementation includes the final specification, implementation and operations-user guide documentation of the day to day activities that ensure high quality operation.

The Implementation-phase, as a consequence of the Conceptual Specification and Binding phases is able to be estimated with great accuracy. Once complete, the effort can be stopped until after all the conversion and deployment activities are accomplished.

The final activity in the Implementation phase is unit, subsystem, and system testing. Because all the specifications, designs, and the like are stored in the Metabase System database, the ability to generate various tests is straightforward. In addition, the Metabase System database, through its user acceptance test specifications, which are founded directly on use-cases, can be employed to ensure comprehensive testing across all the different scenarios.

The next phase is largely indeterminate in length That is because the quality of data that needs to be converted is almost always a disappointing surprise. Additionally, implementation of the database project likely requires additional hardware, network infrastructure, hiring, training, and the thorough testing.

Conversion and Deployment. The Conversion and Deployment phase focuses mainly on the following items: data conversion, equipment acquisition and installation, and training. Since the actual specifications of equipment and networks cannot be realistically known until the Implementation phase is largely complete, current budgets may not be sufficient to handle the acquisition of new equipment.

Included also in this phase is training and, of course, documentation. This is where the effort to fully populate the Metabase System database during these work phases provides significant extra benefit. That is because Metabase System reports can:

- Set out the steps of the use cases including involved processes, database artifacts, and invoked business information systems.

- Identify all the data updates, reports, and business processes related to business events that are related to various business functions across business organizations.

- Prove that database value domains are enforced across multiple data update systems.

- Ensure that all data warehouse reports are based on synchronized data in terms of time, precision, and granularity.

- Provide HR end users support for definitions, business rule descriptions, calendars of business events, Help, and the like.

Production and Administration. The Production and Administration phase occurs only after the HR ERP systems become operational. As an operational system, there are the following major activities:

- Ongoing operations.
- Standard maintenance and evolution.
- Interrogation and evolution.
- Emergency maintenance and evolution.
- Project auditing.

All these activities are affected in a very positive manner through the use of the Metabase System. Not only is the Metabase System database the place where all the work products from the very first project phases are stored, these work products are also continually adjusted and updated as the projects progress across their phases. Consequently, the artifacts that the Production and Administration phase critically depend upon be up-to-date in an end-to-end fashion at the very start of the Production and Administration phase. This enables the very quick identification, analysis, solution formulation, implementation, and testing. During each of the five classes of activities cited above the affected work products are updated as well.

6.2 Applying the Whitemarsh Methodology to the HR ERP Migration Phases

Table 7 presents a cross-reference between the five HR ERP Migration phases and the six Whitemarsh methodology phases. Each cross-referenced cell identifies the Knowledge Worker Framework work products that are created. Table 1 identifies the specific Knowledge Worker Framework work products contained in the row and column cells. The Whitemarsh methodology

sets out the activities that need to be accomplished to produce the work products, and the various Metabase System functional modules capture, store, update and interrelate the work product instances in a non redundant fashion and support their comprehensive reporting.

Each cell consists of an abbreviation of the row's name from the Knowledge Worker Framework and also of the column's name. That is, Scp for Scope, Bus for Business, Sys for System, Tech for Technology, Dep for Deployment, and Ops for Operations. The column abbreviations are: Mis for Mission, DBO for Database Objects, BF for Business Functions, and BO for Business Organizations.

		Whitemarsh Methodology Phases With Knowledge Worker Framework Row & Column Work Product References					
		P1. Preliminary Analysis (PA)	**P2. Conceptual Specification (CS)**	**P3. Binding (Bnd)**	**P4. Implementation (Impl)**	**P5. Conversion & Deployment (C&D)**	**P6. Production & Admin (P&A)**
Metadata and ERP Project Phases	**P1. Build Legacy HR Models**	Scp: Mis, DBO, BF, BO Bus: Mis, DBO, BF, BO	Sys: DBO, BIS				
	P2. Build Enterprise HR Model	Scp: Mis, DBO, BF, BO Bus: Mis, DBO, BF, BO	Sys: DBO, BIS				
	P3. Build ERP HR Model	Scp: Mis, DBO, BF, BO Bus: Mis, DBO, BF, BO	Sys: DBO, BIS				
	P4. Intersect Legacy HR with Enterprise HR Model			Tech: DBO, BIS	Sys: BE, BF, BO Tech: BIS, BE, BF, BO Dep: Mis, DBO, BIS, BE, BF, Org	Dep: Mis, DBO, BIS, BE, BF, BO	Ops: Mis, DBO, BIS, BE, BF, BO
	P5. Intersect Enterprise HR Model with ERP HR			Tech: DBO, BIS	Sys: BE, BF, BO Tech: BIS, BE, BF, BO Dep: Mis, DBO, BIS, BE, BF, Org	Dep: Mis, DBO, BIS, BE, BF, BO	Ops: Mis, DBO, BIS, BE, BF, BO

Table 7. Whitemarsh Phases with Knowledge Worker Framework Row & Column Work Product References

Legend for Table 7	**Rows**	Scope (Scp)	Business (Bus)	System (Sys)	Technology (Tech)	Deployment (Dep)	Operations (Ops)
	Columns	Missions (Mis)	Database Objects (DBO)	Business Information Systems (BIS)	Business Events (BE)	Business Functions (BF)	Business Organizations (BO)

These cell citations, for example, Bus:DBO refers to the following work products from the Knowledge Worker Framework (See Table 1).

- Database Domains.
- Resource Life Cycles.

How the Whitemarsh Methodology phases and first three of the ERP Project phases are brought together is illustrated in Figure 6. The Whitemarsh Methodology phases are identified as columns in Table 7, and the first three phases are identified as the first three rows in Table 7. Figure 6 shows that analysis is accomplished and the resulting work products are stored in the Metabase System database.

6.3 Phase 1: Build Legacy HR Models

The work products that need to be identified, collected, and understood from the legacy environment are identified in the P1. Build Legacy HR Model row from Table 7. Focus is on all but the Business Event column from the Knowledge Worker Framework for both the Scope and Business rows. Built also are the work products from the Database Objects column from the Systems row. The reason this is possible is because these legacy data models already exist–by definition–in order to have operating business information systems and databases. While they may not be elegant they exist.

As shown in Figure 6, the processes necessary for making the discoveries are all provided in the first two of the Whitemarsh methodology phases, Preliminary Analysis and Conceptual Specification. When these work products have been identified, collected, and analyzed, there is a comprehensive "Yourdon Old Physical" that serves as the foundation for both building an Enterprise HR model and also for accomplishing the necessary data integration and continuously running HR data extracts from the various legacy environments to an enterprise-wide HR model.

In many organizations this will be the first time these documents became current since their initial creation during business information system implementation.
These work products are all defined and the processes to capture this metadata are set out in the various Metabase System user guides. The methodology based processes are set out in the Whitemarsh methodology.

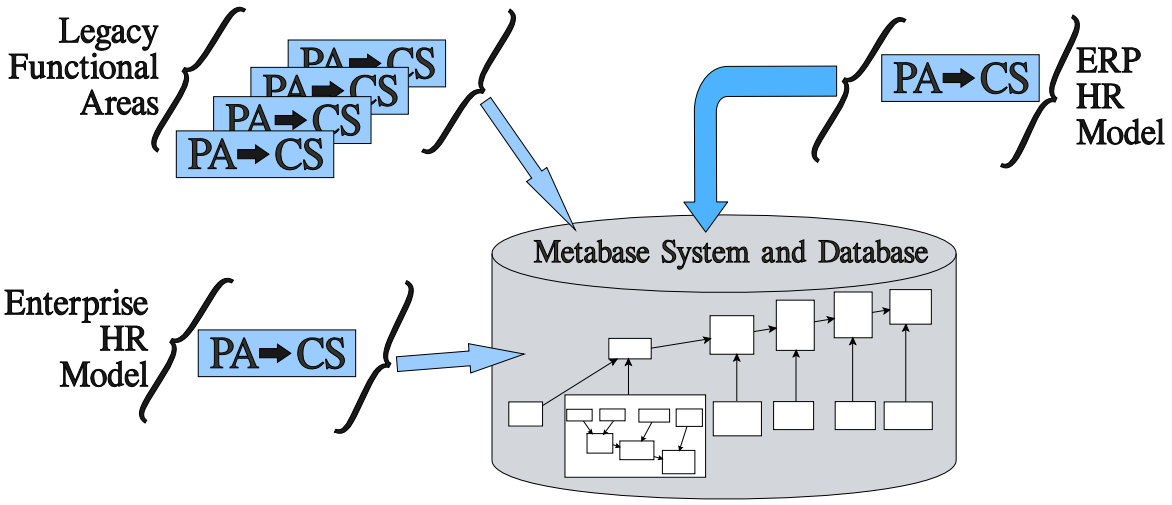

Figure 6. Interrelationship between HR ERP phases 1, 2 & 3 , the Whitemarsh Methodology, and the Metabase System and database.

At the end of this effort there exists a set of "current" data and process models for the entire HR business function in the enterprise. Such knowledge is essential to successfully engineer an enterprise-wide HR model, and ultimately to tune the ERP HR model.

An interesting by-product of these Legacy HR Model activities is a "current" set of documentation for all the legacy databases and business information systems. The completion of this documentation occurs during the Phase 4, which is illustrated in the bottom two rows of Table 7. It should come as no surprise that this documentation is really just a collection of formatted reports from the Metabase System database. These activities are described in the Phase 4 section below.

This legacy documentation work should not be considered optional. That is because a significant quantity of the legacy system and databases are likely to continue and this requires their continued evolution and maintenance. If these Legacy HR business information systems and database documents are throughly documented in the Metabase, this positively affects the ongoing evolution and maintenance of these business information systems and databases. Finally, over time, there may be mergers of legacy functionality as commonalities become more obvious. This approach is therefore a win-win for many reasons.

While tempting, it is both not possible nor desirable that the disconnects among the legacy-based different versions of these HR work products be resolved during this phase. That step is to be accomplished to the maximum extent possible in the next phase.

6.4 Phase 2: Build Enterprise HR Model

The Enterprise HR model is derivable from the previous "Yourdon Old Physical" models developed in Phase 1. The work products created are essentially the same except that they represent an enterprise-view of HR. Such an enterprise view is essential to properly meld data and process semantics from the phase above into the one set that ultimately needs to be stored and employed as the basis for the ERP HR environment.

The various missions are examined from each of the legacy analysis efforts. Determined are the commonalities across the missions. As described in the"Bedford" story (see Section 5.3.1), it is almost always the case that a set of consensus missions can be determined. Specialized missions are often able to be set out as subtypes of a mission supertype. Most of the conflicts disappear when the "Who" and the "How" are factored out and the missions are expressed in their idealized outcome form. The detailed "Who" becomes the work products that describe the different organizations. Similarly, the detailed "How" becomes the different function models.

Similarly, the ability to determine a common set of data models is possible. The commonality starts with determining a universal set of database domains based on common mission hierarchy statements. If there are specialized database domains these, like specialized missions can almost always be represented as subtyped database domains.

During the construction of the Enterprise HR Models, the upper levels of the data semantics are determined. These include data element concepts, data element value domains, and the semantic and data use modifiers that form the foundation of data integration across the enterprise. These Enterprise data models are synchronized with the Legacy modules during Phase 4.

Constructed as well are the enterprise level business function and business organization models across both the scope and business rows.

While there is a great temptation to begin the merger of the Legacy and Enterprise models that temptation should be resisted. That is because there needs to be a complete enterprise-wide perspective for all the work products from the Knowledge Worker columns across the scope and business rows. The method of review must be whether these Enterprise-level models could stand alone without the existence of all the legacy models.

Over time, as commonalities surface among the legacy systems, scope of the consensus-based Enterprise HR model grows. That's progress.

6.5 Phase 3: Build ERP HR Model

The ERP HR model ultimately represents the database and application systems employed to accomplish HR in the enterprise.

The foundation for the ERP implementation is the Enterprise HR model. The first major process of this effort is to identify and catalog the mission, database objects, business functions, and business organizations that implicitly exist within these HR ERP models.

Because ERP HR solutions are to be a solution for all possible HR reasons, the effort to derive a single unified model is a bit of a challenge. The binding of this "divined" ERP HR model to the Enterprise model is accomplished in Phase 5.

A critical element of the formulation is that the enterprise must be able to "see" that its HR functions can be accomplished by the ERP HR solution. For any discovered missing functions, a critical decision has to be made during Phase 5. That is, whether to cause the enhancement of the ERP HR system or to cause this missing functionality to be build into the Enterprise HR Model with a subsequent implementation as standalone functionality.

If at all possible, the decades-long developed and evolved HR traditions and mores should not be cast aside without much soul searching. It must never be forgotten that 50% of all business information system replacements fail after they are deployed due to a lack of user acceptance.

A common technique of the ERP HR vendors is to have a collection of "generic columns" for each table. While this can be a solution to missing functionality, careful considerations has to be paid to whether the time, precision and granularity dimensions can be properly synchronized.

As the work products are discovered, they are entered into the the Metabase System database. The work products are the same except that are created from the viewpoint of the ERP HR package. These represent the "mapped-to" set of ERP HR processes and databases.

6.6 Intersection Phases

With the completion of the three sets of metadata Legacy HR, Enterprise HR, and the ERP HR, the efforts of intersecting can begin. There are two intersections. The first, Phase 4 is the intersection of the Legacy HR Models with the Enterprise HR Model. The second is the intersection of the Enterprise HR Model and the ERP HR Systems. Both intersections are depicted in Figure 7.

The Metabase System database at the start of these two phases contains the work products from the Preliminary Analysis phase and Conceptual Specification phase from all the Legacy HR models, the Enterprise HR model, and the ERP HR model. These work products are shown at a high level in Table 7, at a more detailed level in Table 1, and at a table-named detailed level in the Metabase data Model Diagrams in the Section 10 referenced Metabase Overview document. The most detailed level of all is the actual Metabase Database SQL DDL that contains the full table, columns, and relationship specifications. These come as part of every Metabase System installation.

The products of these next two phases are the business information systems and databases that comprise the end-to-end ERP HR environment. The Whitemarsh methodology

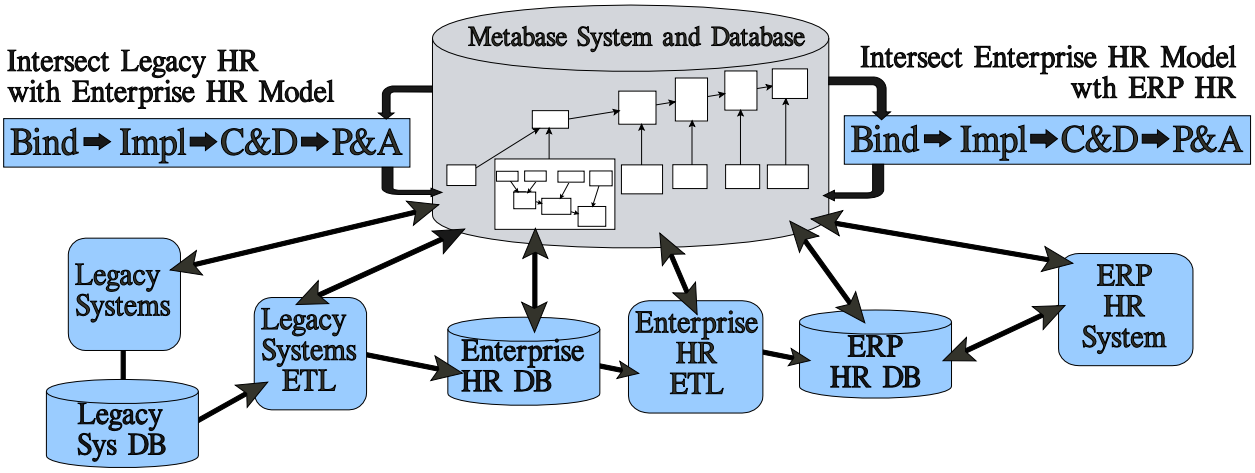

Figure 7. Interrelationship between HR ERP phases 4 & 5 , the Whitemarsh Methodology, and the Metabase System and database.

phases for both these HR ERP Migration Intersection phases are Binding (Bind), Implementation (Impl), Conversion and Deployment (C&D), and Production and Administration (P&A). The key steps and the work products for each HR ERP Migration Intersection phase is described in terms of the Whitemarsh Methodology phases.

6.7 Phase 4: Intersect Legacy HR with Enterprise HR Model

With the completion of the three sets of metadata, that is, Legacy HR, Enterprise HR, and the ERP HR, the intersecting efforts can begin. Intersecting the work products from Phase 1 involves these distinct processes:

- Determining a generalization for a collection of legacy work products.
- Recording the mapping of a legacy work product with one or more Enterprise HR work product.
- Recording the compromises that may have to be made when the intent of a legacy work product is accomplished by an Enterprise HR work product.
- Development of a complete set of production-ready ETL business information systems and databases to accomplish a Legacy to Enterprise HR operating environment.

Once an intersection is accomplished, it is prototyped to determine if it is reasonably acceptable to each legacy organization. These prototypes involve taking the HR Enterprise data model and creating a data model and process model infrastructure for accomplishing the data migration from the legacy HR component and the Enterprise HR component.

At each iteration of a Legacy to Enterprise HR prototype, the required changes to the work products are stored in the Metabase System database. This enables a constant set of currentness across all the work products with the legacy HR metadata, the Enterprise HR model, and between the specifications contained in the legacy and the Enterprise HR Model specifications.

From Figure 7, the four items that are addressed are:

- Legacy Systems.
- Legacy System Databases.
- Legacy Systems ETL.
- Enterprise HR Database.

Binding. The key steps for Binding ensure that there is a complete ETL migration path for all the data extraction systems that need to be created to obtain data from the various Legacy business information systems and databases and the storage of that extracted data into the Enterprise HR databases.

The role of the Metabase System database is to capture all the mappings from the Legacy HR models to the Enterprise HR models. These mappings center on value domain mappings, reconciliation of time, granularity and precision.

Legacy Systems and databases may need to be modified to ensure consensus across all enterprise shared data.

A key architecture decision is whether there are intermediate data file structures that represent all the shared HR data and whether these data structures result from an automatic "push" from the legacy systems or are the result of scheduled "pulls." This paper take no position on that architecture decision. Another architecture decision is whether all the extracted data that is stored in intermediate file structures is in a "database" form or an XML form. If the former, there has to be a tighter binding between the database structure of the Legacy databases and that of the Enterprise HR model databases. If XML, the intermediate data transactions can be both more generalized and in a non-technology bound form.

Regardless of these architecture decisions, these mapping activities may cause changes in the Physical Database component of the work products created during the Conceptual Specification phases. Because of these changes, there is a both a "read" and a "write" arrow from the Metabase System and database and the graphic depiction of the four Whitemarsh phases (Bind, etc) in Figure 7.

Multiple DBMSs can be an issue addressed during the Binding phase. For example, some of the Legacy applications may be Microsoft Access databases, others may be MySQL, and still others may be Oracle, or DB2. A multiple DBMS environment may impact the ETL processes

because the extracts have to come from one DBMS, stored in intermediate files in a second, and possibly store the transformed data through a third DBMS. This complicates the overall environment that must be properly technologically bound prior to the Implementation phase.

During the binding phase, the prototypes created during the Conceptual Specification phase need to be enhanced to support ETL between Legacy and the Enterprise HR Model. By the end of the Binding phase all the architecture issues should be addressed, all the functionality that needs to be turned into production systems should be prototyped. The databases for both Legacy and Enterprise HR should be completed and successfully compiled in the appropriate DBMS.

Finally, all the work products that were accomplished during Preliminary Analysis, Conceptual Specification, and Binding should be completely updated within the Metabase System database.

Implementation. The key steps for Implementation are the development of the ETL systems that extracts data from the legacy systems and store the data into the Enterprise HR databases. The activities for Implementation are rather traditional. That is, Logical, Physical, Interrogation, and System Control.

Logical includes:

- Complete database schema final adjustments for both the extracting Legacy databases and the receiving Enterprise HR databases.
- Finalization of all data value mappings.
- Finalization of all data model designs addressing time, granularity and precision

Physical includes:

- Compilation and testing of all schemas including in this would be any necessary XML and/or SQL View specifications.
- Detailed mapping between the Legacy databases and the Enterprise HR database models.
- Test data development.
- Specification, development, testing, and documentation of all software systems for extraction, transformation, and loading.
- Development and testing of all business event and calendar based schedules for executing the ETL.

- Creation of necessary reports that provide event or calendar-based information regarding the ETL.

Interrogation includes reports necessary to provide event or calendar-based information regarding the ETL.

System Control includes careful attention to:

- Audit trails
- Backup and recovery
- Concurrent operations
- Security and privacy
- Message processing
- Logical and physical database reorganization
- Multi-database processing
- Installation and maintenance
- Application optimization

Each area needs to be designed, implemented, tested and incorporated into the operations schedules for the ETL systems between Legacy and the Enterprise HR databases.

Two more areas are addressed within the Implementation phase: Ancillary Supports, and System Quality Tests.

Ancillary supports that are developed and set into place include: database and application software training, hotline, standards update for the database and application software, test data, and documentation that includes database and application software document. With respect to documentation, these materials are generated from the Metabase System database.

System Quality Tests are a follow-on from the unit testing that occurs for each database, ETL, and various application support modules. The end-user facing system quality tests are based on the use-cases that are developed during System and Business Information System cell set of activities that have during Phase 1, Build Legacy HR Models.

Within Implementation, these tests are generated and reviewed by the functional experts to ensure that they are both comprehensive and highly discriminating. In addition to end-user facing tests, comprehensive system quality tests must be developed and exercised against the created business event and calendar schedules. A necessary part of the system quality tests is a thorough auditing of the test data and of the results to ensure that the tests are sufficient, embrace all known edge cases, and produce predetermined correct results.

Conversion and Deployment. Conversion and Deployment presumes that all the activities within the Implementation have been completed. Except for the activities within the Conversion and Deployment, itself, the databases and application software are 100% ready for production at the start of the Conversion and Deployment.

The key activities included within the Conversion and Deployment are:

- Data conversion.
- Hardware selection, installation, and testing.
- Network support selection, installation and testing.
- Complete end-user training.
- Standing up the hot-line for end-users.
- Deployment of all databases and application software to production hardware.
- Deployment of all necessary backup and recovery facilities.

Production and Administration. The key activities for the Production and Administration are:

- Production system operations that include daily data acquisition and updating, and scheduled event-based processing for reports.

- Ongoing development of interrogation facilities for operational, tactical, and strategic reporting. For the tactical and strategic reporting the most likely source of data is either daily or weekly data extracts that ensure stability of time, granularity, and precision.

- Application optimization activities that provide automatic reaction to continuous performance testing of the efficiency of data loading, updating, reporting, and system control concerns including for example, audit trails, backup and recovery, message processing, and security and privacy.

- Development of standardized processes for system evolution and maintenance.

- Execution of emergency maintenance and then followup adjustment of all Metabase System database specifications of work products.

- Regular evolution and maintenance.

- Comprehensive auditing, quality assessments, and development of lessons learned.

6.8 Phase 5: Intersect Enterprise HR Model with ERP HR

The process of intersecting the Enterprise HR model with the ERP HR model is simpler because there is only one model for each. As stated at the outset of this paper, the reason for an Enterprise HR model are mainly to:

- Enable a unified model for all key HR data and processes across the enterprise.
- Maintain an independence between the enterprise and the ERP HR vendor.

At the end of this second intersection effort, not only are a complete set of specifications for accomplishing legacy to ERP HR migration developed, so too are the ERP HR business information systems and databases. The first intersection effort accomplishes:

- he development of an Enterprise HR model.

- The mapping of the diverse legacy applications and databases to a unified Enterprise HR database model.

- The development of a complete set of specifications that document the effort.

- The development of production ready ETL systems that provide data to the ERP HR environment to support a successful enterprise-level HR environment.

The second intersection effort accomplishes the development of the mappings between the enterprise HR model and the model that has been created by the HR ERP vendor. In addition, the prototypes developed during the second intersection effort enables the demonstration of the actual operation of the ERP HR system and to understand how the legacy applications and databases are served by the operational ERP HR environment.

In addition to these specifications, prototypes of the ERP HR model exist and can be approved, disapproved, and/or modified by the enterprise HR community. These prototypes differ from the first set because they do not mirror just an unimplemented enterprise-view of HR but are a prototype-based implementation of the actual ERP HR implementation.

At each iteration of mapping Enterprise HR model prototypes to ERP HR prototypes, the required changes to the work products need to be stored in the Metabase System database. This enables a constant currentness within and across all the work products. Once the Phase 5 set of prototypes are completed, there are:

- The legacy HR metadata.
- The Enterprise HR model metadata.
- The intersection metadata between the legacy and Enterprise HR models.
- The ERP HR metadata.
- The intersection metadata between the Enterprise HR model and the ERP HR model.

In short, a comprehensive and complete specification of what is implemented. These specifications are thoroughly prototyped to ensure its validity.

The activities within the Binding, Implementation, Conversion and Deployment, and Production phases are largely the same as those contained in Phase 5. Hence they are not be repeated here. The focus of these activities changes as they address two areas:

- ETL from the Enterprise HR databases to the ERP databases.
- End-user access to the ERP HR environment for various business purposes.

The first is obvious and is not further elaborated. The second, however is a bit more complicated. An architecture decision has to be made early on. That is, whether all end-user facing processes are perceived as coming from1) the Enterprise HR environment, 2) the legacy HR systems or 3) exclusively from the ERP HR environment, or a hybrid of the three.

Under the first alterative, the ERP HR system and its database are really just an "access method" through which data is selected, formulated into XML streams, and is loaded into the Enterprise HR database for end-user presentation through the Enterprise-HR layer.

Under the second alternative, the end-user facing data is provided back to the legacy applications for presentation to the legacy-based users.

Under the third alternative the end-users directly connect to the ERP HR environment for all their HR data.

The factors that would bear on the choice are:

- The quantity and type of legacy data and processes that need to be preserved to the legacy users along with the inclusion of enterprise data and also ERP HR based data. This architecture requires changes to the legacy business information systems and databases.

- The quantity and type of enterprise data to be commonly formatted and presented to end-users separate from legacy data and processes. This architecture preserves the legacy business systems to the greatest degree.

- The quantity and type of ERP HR data to be provided to all end-users. This architecture alternative minimizes or eliminates changes to the both the legacy business information systems and database and the Enterprise database environment. In fact, this alternative might eliminate the Legacy HR and Enterprise HR and their implementations altogether. However, this results in Alternative 1 that was described in Section 5.1 that was discarded at the outset.

These three alternatives significantly affect the return-trip ETL activities from the ERP HR environment to the Enterprise HR environment and in turn, from the Enterprise HR environment to the legacy business information system and database environments. If all the end user presentation is from the ERP HR system, the Legacy HR and the Enterprise HR models can be

eliminated. That however makes your enterprise HR functions, processes and data entirely captive to the HR ERP vendor.

If all the end user presentation layer is from the intermediate Enterprise HR layer, then while the ETL from the HR ERP to the Enterprise HR must occur, the enterprise is largely independent from the HR ERP vendor, and as such, ERP vendor replacement is possible.

If all the end user presentation is from the Legacy HR systems, the need for Enterprise HR to Legacy HR ETL is significant. Preserved, but in a negative fashion will be the current unique and possibly culturally conflicting legacy HR systems that currently exist across the enterprise.

For all these reasons, the second end user presentation layer choice, which maps to the Section 5.3 chosen alternative continues to be the recommended choice.

7.0 Enterprise Information Technology Intellectual Property Management

The description of the scenario for migrating HR Legacy business information systems and databases to an HR ERP is largely complete. It is now important to reflect on the processes and domain of the metadata collected in this HR ERP Migration scenario that can support other Enterprise IT scenarios. The overall HR ERP required metadata domain is illustrated in Figure 8.

The steps contributing to an understanding of enterprise-wide metadata applicability are:

- Understanding the processes and models associated with the manufacturing work products and their integration into one overall set of enterprise intellectual property.
- Showing how the development of all these work products is accomplished though the varied staff in different organizations.
- Setting out the strategy for integrating all the work products into one overall enterprise-level set of intellectual property.
- Identifying the applicable enterprise IT scenarios that are generally similar to the HR ERP migration effort.
- Setting out the advantages and benefits derived from an enterprise wide set of IT intellectual property.
- Indicating a way ahead for this type of effort.
- Showing why the overall IT environment must change fron one that is project-centric to release-centric.

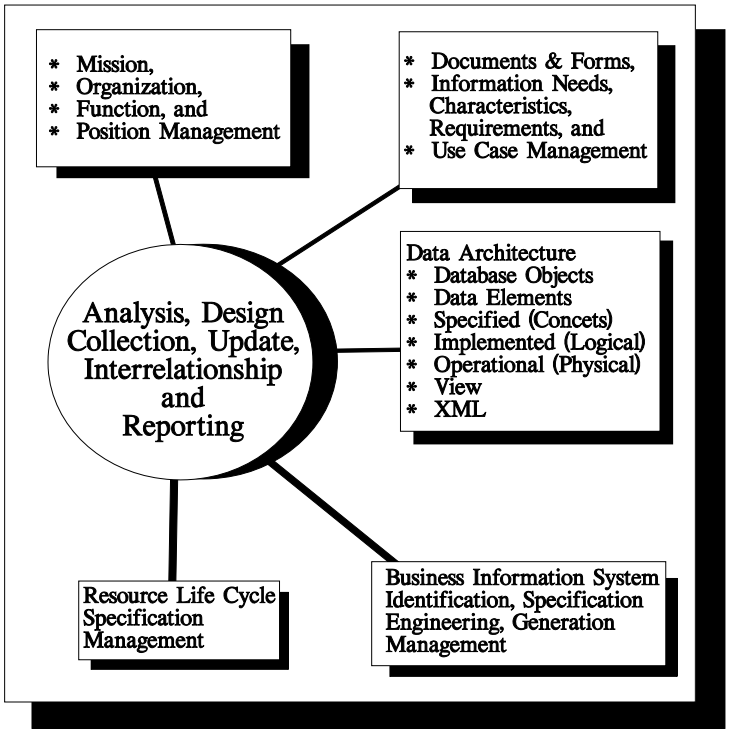

Figure 8. Metadata domain for HR ERP migration.

7.1 Work Product Manufacturing and Integration

Figure 8 summarizes the collection of work products that exist across the Knowledge Worker Framework. The various metadata data model diagrams from the Metabase Overview document detail the engineering of each of the work products. It, at first, may seem overwhelming. How to understand it all? How to accomplish it all? Isn't it more work than necessary?

Initially that may appear to be true, but there are several integrating mechanisms that makes it simpler, less work, and a smaller volume of work products. First is the Whitemarsh methodology that is squarely based on long-standing, traditional work products that set out what has to be done, in what sequence, and "where" it goes. Second, the Metabase System is engineered to maximize the use of prior work. Third, the supporting documentation, books, courses and seminars that set all this down in a logical sequence. Finally, this is just highly-proceduralized common sense. A common exclamation from IT professionals after it is explained is simply, "Well, duh!"

Key to enterprise integration efforts is data architecture. It plays a critical, interlinking role across all the work products. In the Section 10 referenced document, *Challenge to Business Information System Success,* Figure 4 and Tables 8 and 9 show how the various contained data

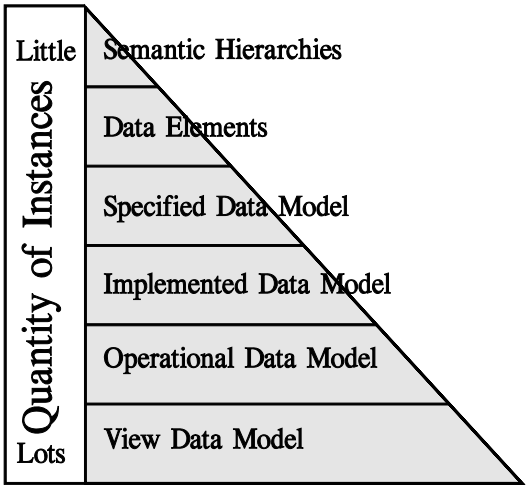

Figure 9. Quantity distribution of metadata across data models within the enterprise data architecture.

models within an enterprise's data architecture interact with virtually all the other work products cited in Figure 8.

The collection of metadata across all these data models might appear to be massive. Actually, it is not. For sure there are as many operational data models as there are actual databases. But across the higher level abstractions, there is less and less metadata. That is because of its define-once and use many-times architecture.

Figure 9 illustrates the following increasing-volume hierarchy:

- Semantic taxonomies are defined once but are employed multiple times to brand business facts (business data elements, attributes of entities, and columns of tables) in support of automatic (yes, automatic) naming and definitions.

- Data Elements, which are database independent business fact semantic templates, are defined once but are employed many times to define the semantics infused into attributes of entities and columns of tables.

- Specified Data Models, which are data models of individual concepts, are defined once per concept, and are employed many times to standardize business fact collections within and across implemented (i.e., logical) data model tables.

- Implemented Data Models, which are logical database schema models that are independent of DBMS, are defined once per database and serve as the foundational basis for one or more operational (physical) data models. Because an Implemented Data Model table can be constructed from more than one Specified Data Model entity, and also be employed within multiple Implemented Data Model tables of the same or different schemas, the relationship between the Specified Data model and the Implemented Data Model is many-to-many.

- Operational Data Models, which are physical database schemas tied to specific unique combinations of DBMS, hardware platforms and operating systems, are typically engineered to achieve specific physical performances. Operational Data Models may, in the case of data warehouses, be the union of multiple Implemented Data Model table collections. Consequently, the relationship between the Implemented Data Model and the Operational Data Model is also many-to-many.

- View data models, which are the intersection of a DBMS-based data model and a business information system, are defined once and used many times to represent these intersections.

Because of all this "define-once, use many-time" engineering, there is an increased level of "automation and manufacturing" versus individual, redundant, one-off designs. Not only is each layer integrated with its higher layer in a top-down fashion, this approach also supports the inductive development of higher layers through reverse-engineering of lower layers.

Figure 10 illustrates the define-once use many times integration of the Data Element Model with the Specified Data Model. The data element model is illustrated in the right side. There is the data-based concept, Address, which, in turn, includes the data element concept, Postal Address. That, in turn, contains the business data element, Postal Address City, which becomes the semantic template for City in three different entities: Person Education, Person Biography, and Person Address. In each is a historical fact, School City, Birth City, and [residential] City. What is branded are the semantics from Postal Address City. That includes data type, length, and possibly value domains

When a Specified Data Model of a concept is created, for example, Person Biography, if it involves the semantic specification of a city, the business data element's semantics for Postal Address City are branded into the City attribute.

Similarly, Figure 11 illustrates the define-once use many-times relationship between the Specified Data Model and the Implemented Data Model. In this example, there is the concept, Person Address. In the Implemented Data Model, there are two tables, Customer and Sales Region. In both of these tables there is an address.

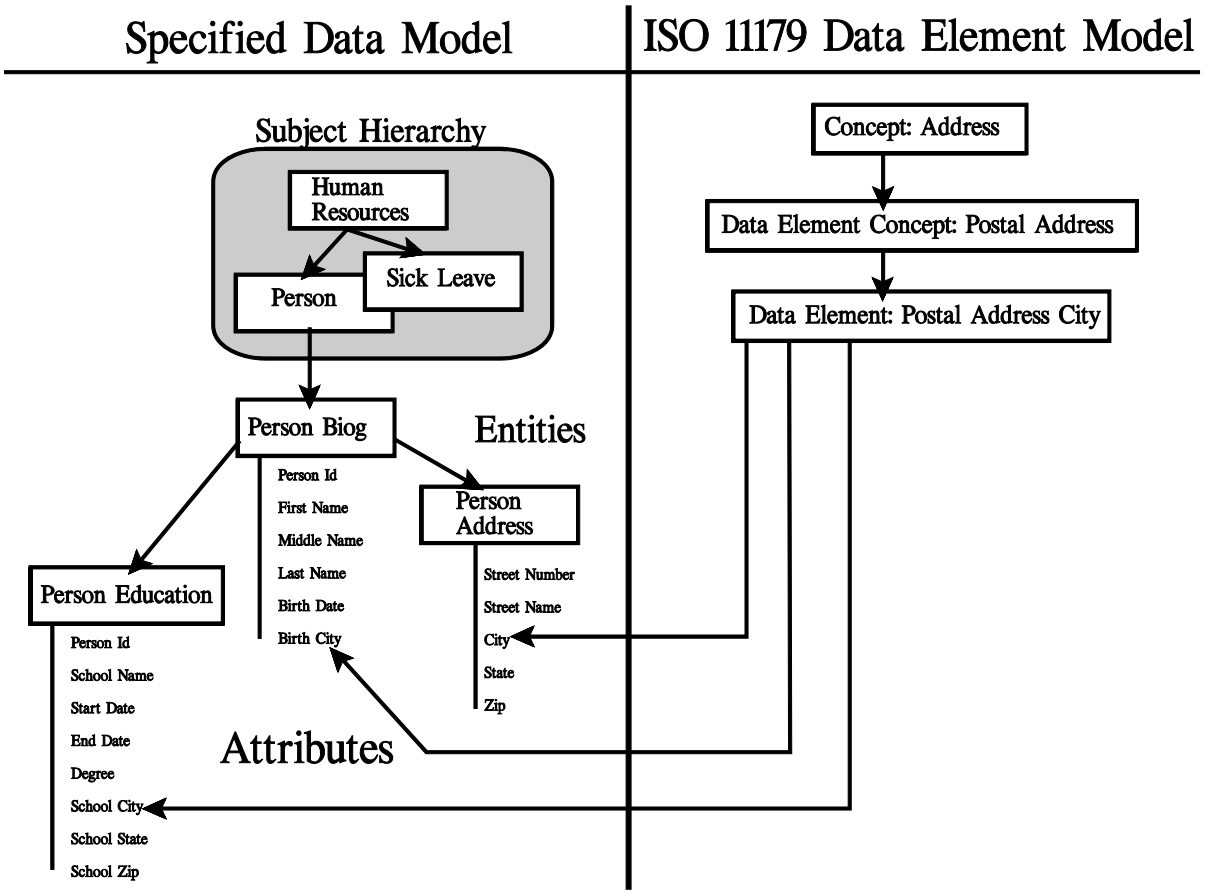

Figure 10. Define-once, use many-times integration between the data element model and the specified data model.

During the construction of the Implemented Data Model tables, the Specified data Model concept, Address, is imported to participate in the table and its columns. It is common that multiple entities from Specified Data Model concept models are employed in the construction of the same Implemented Data Model table.

For each such participation, a backwards relationship results between the Implemented Data Model and the Specified Data Model. There is also a relationship to the Data Element Model through the related Specified Data Model. This enables the reporting of the use of business data elements employed in the Implemented Data Models across all the Implemented Data Model databases.

Figure 12 illustrates the define-once, use many-times relationship between an Implemented Data Model and an Operational Data Model. In the case of a data warehouse, its data model is almost always the use of tables from multiple Implemented Data Models. In this specific example, the Operational Data Model contains the DBMS table, Sales. It is the

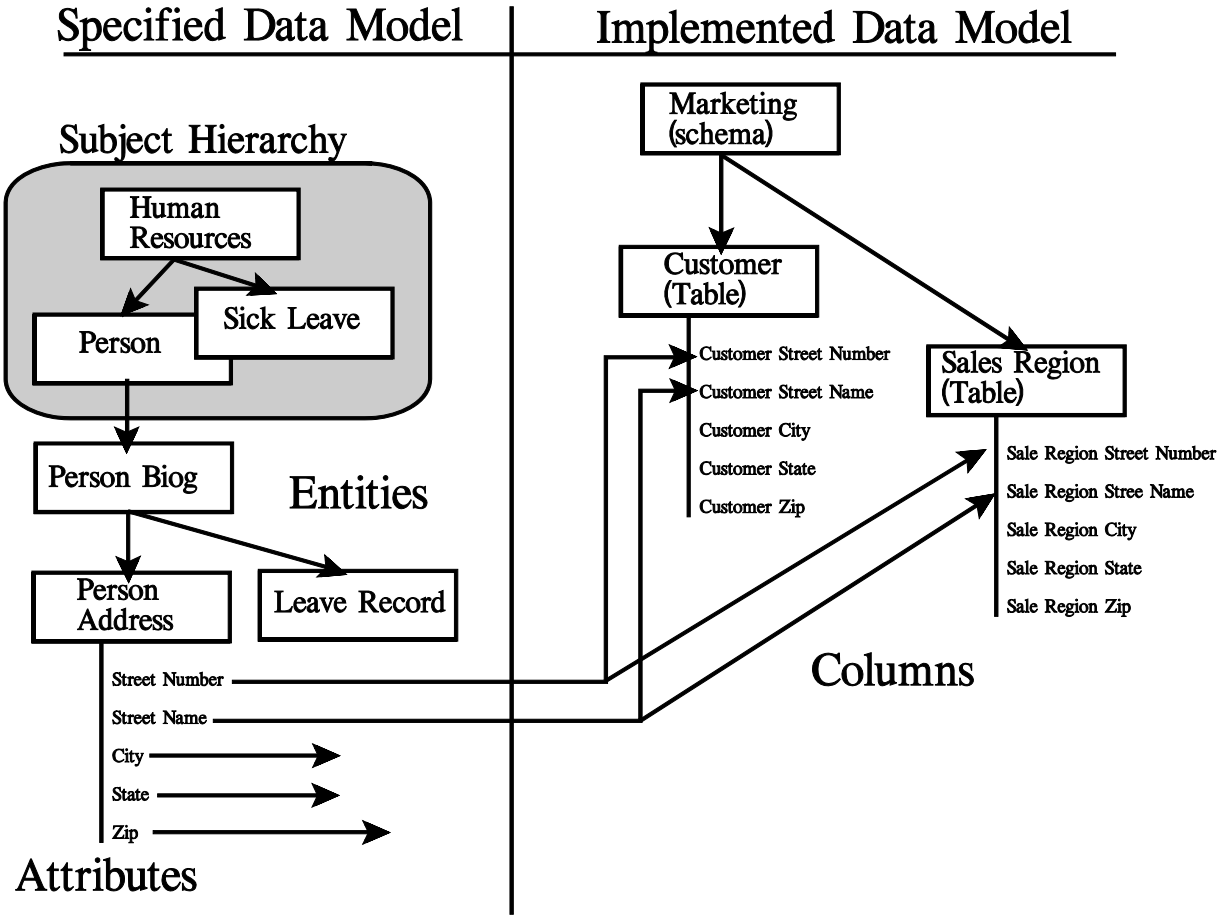

Figure 11. Define-once, use many-times integration between the Specified Data Model and the Implemented Data Model.

intersection of the Implemented Data Model table, Sales and the Product table that was not in the Marketing schema of the Implemented Data Model. It is from an entirely different Implemented Data Model schema. Thus, there is a many-to-one relationship between an Implemented Data Model table and an Operational Data Model DBMS-table.

Figure 12 illustrates the define-once, use many-times relationship between an Implemented Data Model and an Operational Data Model. In the case of a data warehouse, its data model is almost always the use of tables from multiple Implemented Data Models. In this specific example, the Operational Data Model contains the DBMS table, Sales. It is the intersection of the Implemented Data Model table, Sales and the Product table that was not in the Marketing schema of the Implemented Data Model. It is from an entirely different Implemented Data Model schema. Thus, there is a many-to-one relationship between an Implemented Data Model table and an Operational Data Model DBMS-table.

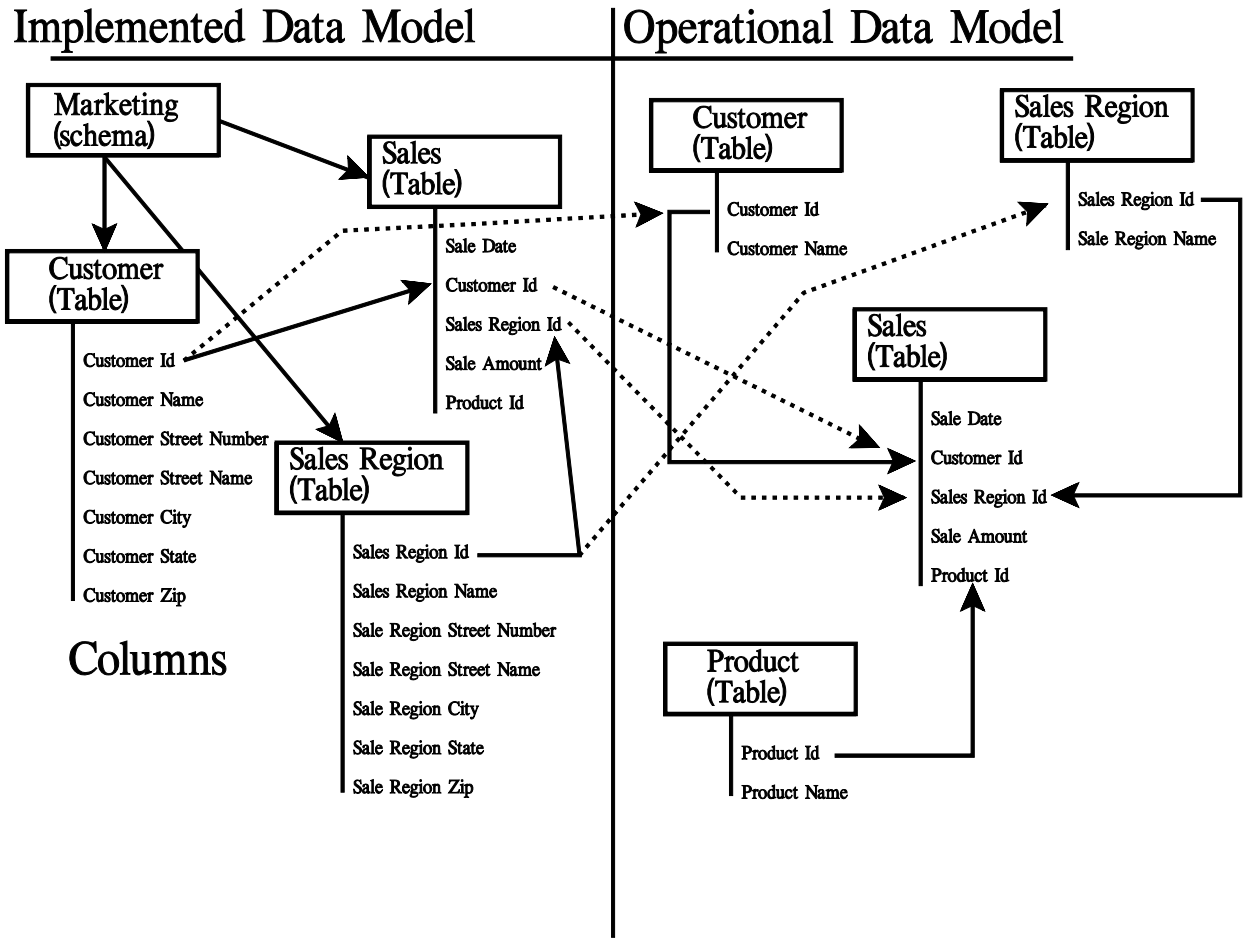

Figure 12. Define-once, use many-times integration between the Implemented Data Model and the Operational Data Model.

There is almost always a one-to-many relationship between and Implemented Data Model and a collection of Operational Data Models wherein each Operational Data Model is designed to achieve a unique performance objective. Overall then, there is a many-to-many relationship between the Implemented Data Model and the Operational Data Model.

Imported here are three Implemented Data Model tables, Customer, Sales, and Sales Region. Once imported, a number of columns in these Operational Data Model tables were deleted because the created Operational Data Model database is part of a sales reporting data warehouse database.

As with the prior example, there is a backwards relationship between the Operational Data Model and the Implemented Data Model. There is also a relationship to the Data Element Model through the related Implemented and Specified data models. This enables the reporting of

business data elements employed in the Operational Data Models across all the Implemented Data Model databases.

As a consequence of all this define-once, use many-times effort, a number of data standardization and work saving benefits result:

- The semantics are able to be associated with business data elements, attributes and columns in a hierarchical manner that are semantically more narrow as they proceed down through the data model layers.

- Business fact names are automatically constructed as well as be arbitrarily set by the data modeler. Both are maintained by the Metabase System.

- Business fact definitions are able to be automatically constructed that include a local definition fragment that is subsequently enhanced through the higher layers of assigned business fact names and definitions and entity/subject/table/schema contexts.

- Comprehensive reporting is supported starting at the highest levels so that where-used reports are able to be shown. SQL DDL is able to be generated at the Specified, Implemented, and Operational Data Model levels. Data modeling tools that display data model diagrams such as DeZign from Datanamic.com can be employed to import SQL scripts generated by the Metabase System.

7.2 Work Effort Distribution

Figure 13 illustrates how data architecture work product creation can be distributed across the enterprise. The top area, “central data semantics and data element standardization” would be accomplished by Data Architects. The Specified Data Models of concepts would be created by Functional Data Administrators, and the Implemented, Operational, and View Data models would be accomplished by traditional IT staff for data administration, database administration, and business information systems development.

While this diagram only shows data models, similar work product development distributions are accomplished for all the Knowledge Worker Framework work products.

For example, from Figure 8, the Mission, Organization, et al collection of work products as well as the Resource Life Cycle work products are created by Enterprise Solution Architects. The Documents, Forms, Information Needs, et. al collection of work products are created by Business Analysts. The Data Architecture work product collection is created by data architects and modelers. The Business Information System work product collection is created by traditional IT development staff.

What makes the distribution of work products possible is the multi-user, Internet accessible Metabase system. Its database design is explicit, and is supported by SQL engines.

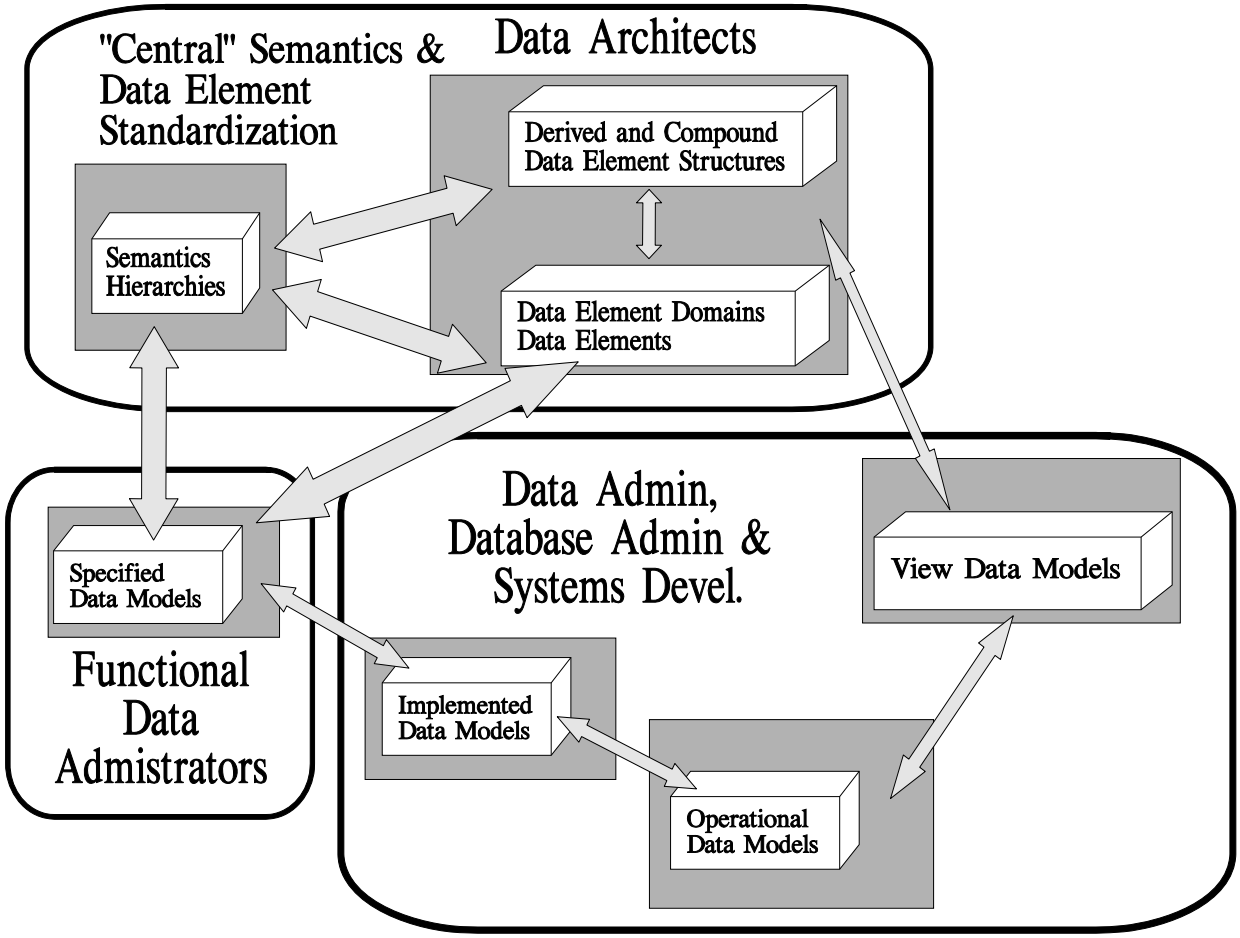

Figure 13. Distribution of Data Architecture work across the enterprise.

These last two characteristics make reporting from the Metabase database easy as any ODBC supported report writer can be used to quickly develop reports.

7.3 Enterprise Intellectual Property Integration

Table 1 presents the Knowledge Worker Framework that sets out the work products across major categories (Mission, Database Objects, etc.) in varied levels of detail (Scope, Business, etc.). This framework of what needs to be done is independent of methodology, business functional area, and Enterprise IT scenario.

Within this 6x6 framework are twelve IT-centric cells. These IT cells are shaded and appear in Table 2. Essentially, while these 12 cells are the responsibility of IT, they contribute

less than 10% to IT's overall success. Thus, 90% or more of the reasons why IT succeeds or fails occur within the surrounding 24 Knowledge Worker Framework cells. It would be nice if IT only got 10% of the blame when failures occur. We know, however that IT is commonly cited as the 100% source of failure. It is therefore incumbent on IT to embrace, facilitate, and take responsibility for the entire set of work products across all 36 cells.

IT needs to set out the projects, methodologies, products specifications, and accomplish work product *integration, interoperability, and non redundancy* (there's that three part phase again) across all enterprise functional areas and organizations. In short, one set of enterprise intellectual property across all reasons.

Table 6 sets out the six phrases (preliminary analysis, conceptual specifications, etc.) of the Whitemarsh methodology Within these phases are the process descriptions for creating the work products. Figure 6 of clearly implies that work can be done in parallel across the legacy functional areas, and for the ERP's HR model. Once the legacy functional areas work products are developed, the Enterprise HR model can be created. Of course, all the developed work products are stored in an <repeat that three part phase here> manner.

Table 7 sets out the intersection of the Whitemarsh methodology (columns) and the ERP project functional areas (rows). The cells are the short-hand abbreviations of the developed work products.

Keeping all this together is possible for two key reasons:

- The Whitemarsh methodology is independent from business functional area influence.

- The project management module of the metabase system defines, collects, reports and operates on "project data," which is just another type of metadata.

The Whitemarsh Short Paper, Manufacturing Project Plans, referenced in Section 10 explains this is greater detail.

Figure 14 depicts the intersection of:

- Knowledge Worker Framework columns (and implied rows for specificity)
- Whitemarsh Methodology Phases
- Functional Project Phases (e.g., the HR ERP effort)

Figure 14 shows that during Phase 3 (Ph-3) of the functional project, a work product package for database objects (DBO) was created during the Conceptual Specification (CS) Phase. Table 7 indicates that the database object specifications support P3 Building the ERP HR Model. Table 1, show that the construction of the work products are at the intersection of the Database Objects column and the System row. That is, Data Elements Models, Specified Data Models (data models of concepts), and the identification of Database Objects.

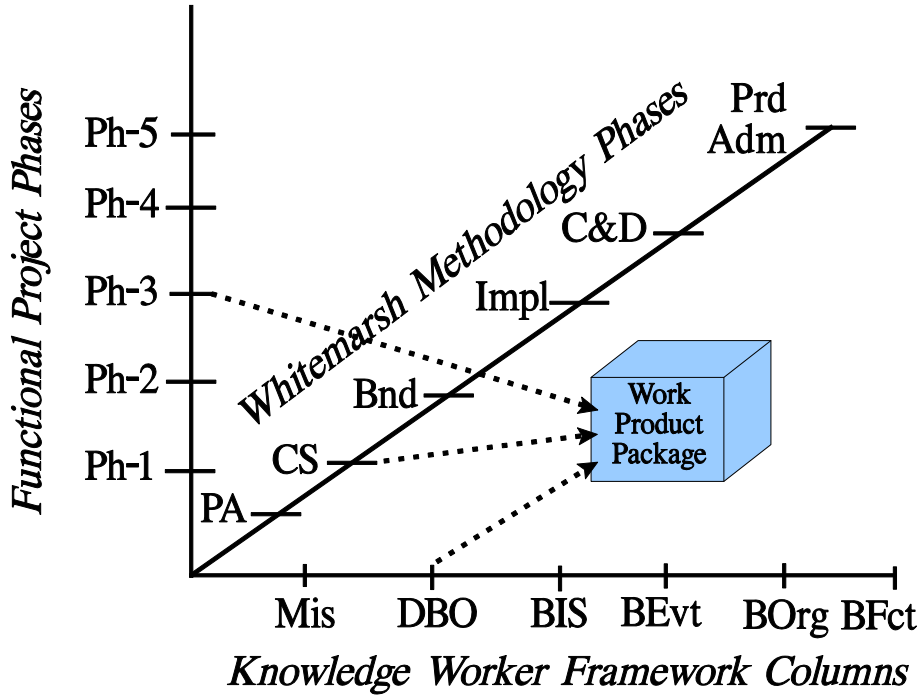

Figure 14. Functional Project Specific Work Product Package across Whitemarsh Methodology and Knowledge Worker Framework.

Now, if there was just one enterprise functional project, the Metabase database would be filled with ERP work product package metadata that would be <insert the three part phase here>. If that were done across each of the Enterprise Intellectual Property Scenarios, then, while there would be a dramatic improvement in contrast to having just requirements in a requirements modeling tool, data models in a data modeling tool, process models in a process modeling tool, and project plans in a project management package, there would still be independently developed, stove-piped collections of work products that are not integrated, not interoperable, and not non-redundant across the enterprise.

In this specific example, the work package would contain products for the Data Element Model, Specified Data Models (data models of concepts), and the identification of Database Objects. As this ERP project proceeds across the Functional Project Phases, these metadata work products become the basis for other work products, are evolved to incorporate changes, and ultimately represent one set of intellectual property across the entire HR ERP effort.

Because the Metabase System is just a metadata-oriented data-centric system, Figure 15 is operative.Through this strategy, the individual work packages, developed in an integrated manner across the various phases of one Enterprise Intellectual Property scenario area, for example, HR ERP, are themselves integrated and stored in the metabase system's database with all the other work products from other Enterprise Intellectual Property scenarios such as

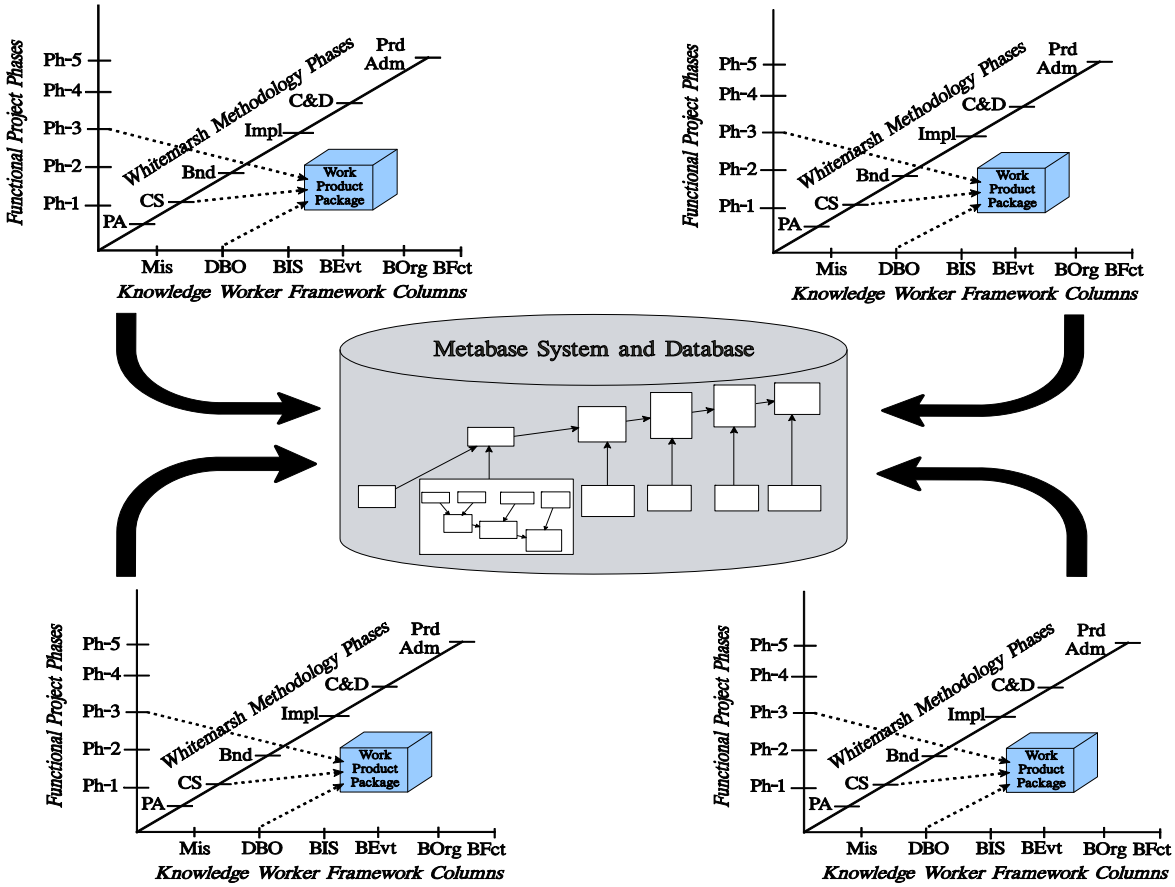

Figure 15. Work package integration, interoperability, and non redundancy across all enterprise functional areas.

Business Information System Integration, or Enterprise Architecture such that only one set of metadata results. Thus there would be only one enterprise-wide Data Element Model, one collection of Specified Data Models (data models of concepts), and one overall set of enterprise Database Objects. In this example, there would be multiple Implemented and Operational Data Models as these are often business function specific.

7.4 Applicable Enterprise Information Technology Scenarios

Five scenarios easily come to mind that have immediate applicability to the strategy contained in this paper. Each is representable through collections of overlapping metadata, work products, and Whitemarsh methodology tasks. Much of their content is the very same. It is just that the purpose for the content is different.

The scenarios, described in Table 8 are

- Migration to ERP
- Business Information System Integration
- RFP Development
- Enterprise Architecture
- Data Management Engineering and Governance
- IT Project Intellectual Property preservation

Scenario	Description
Migration to ERP	The content of this paper.
Business Information System Integration	The melding of disparate implementations of functionally similar business information systems that may be from different vendors with similar but different collections of database tables.
RFP Development	The creation of a complete specification that enables vendors to bid on an IT project. Included would be the Mission, Organization, and Function models. Included as well is a complete data architecture reference model, and finally a complete set of working prototypes that ensure that the vendors have a good handle on requirements.
Enterprise Architecture	The creation of a large collection of metadata across all existing and operational business information systems that can serve as the foundation upon which all future development and maintenance occurs.
Data Management Engineering and Governance	Identification, collection, and integration at the Semantics, Data Element and Specified Data Models of the enterprise collection of Implemented and Operational Data Models. Integration of multiple, but similar Operational Data Models into a reduced set of Implemented Data Models. Finally, the administration of this enterprise-wid data architecture through governance processes that ensure consistency, reliability, and repeatability.
IT Project IP Preservation	Because of the increased use of ERP systems, COTS application packages, the development of Business Information Systems through contractors, and finally of the aging of internal IT staff, there is a severely reduced information technology corporate memory. It is sadly common for “nobody” to have a clue as to what is wrong, where it is wrong, or how to fix it. This must be reversed. The Metabase System and its database is definitely a way to store, preserve, and evolve extracted corporate memory--independently--from those who created it.

Table 8. Enterprise Intellectual Property Scenarios.

7.5 Advantages and Benefits of Enterprise Information Technology Intellectual Property Management

This approach has a number of advantages and benefits including:

Divide and Conquer. By accomplishing the identification of the legacy HR processes organization by organization, the overall effort is faster. Additionally, it enables the "tried and true" best processes, data structures, and practices to surface and be recorded without "editorial judgement" or any attempt to meld them into an enterprise best practice. This occurs later. As the scenarios identified in Section 7.4 are accomplished either serially or in parallel, a foundation of enterprise-wide metadata is developed and loaded into the Metabase System and database. This leads to the acceleration of individual efforts because commonly needed work products will have already been created.

Metadata Recording and Integration. All the work products are really metadata. By gathering and recording them in an integrated, interoperable, and non redundant manner, the overall effort is faster as multiple work products that are the same only have to be captured once while being mapped multiple times. In short, the use of these metadata recording techniques increases productivity and quality while reducing cost and risk.

Parallel Phases. As illustrated in thie Enterprise HR ERP migration scenario, the first three ERP HR functional phases can be accomplished in parallel (See Figure 6). The work products of each are the same type but different content. Because these work products are of the same type, and because there are intersecting processes, the ability to intersect the work products of Phases 1 and 2 together and those of Phases 2 and 3 together enable a single unified set of work products that are able to be integrated, interoperable, and non redundant. This same parallel work accomplishment characteristic exists in all the other Enterprise IT scenarios.

Been There, Done That Approach. The Metabase System is system supported by books, courses, methodologies, metrics, and seminars has been accomplished multiple times since the middle 1980s and have show dramatic reductions in cost and time, and significant increases in productivity and quality. After each occurrence, the "hot wash" causes improvements. Additionally, detailed examinations of the work products' engineering and integration coupled with the common-sense methodology that is engineered to be accomplished in pieces, cycles, and by multiple common-purpose teams lends itself well to this type of effort.

Enables a Way Ahead for Future Enterprise Projects. The work products from the three main ERP HR phases, the two ERP HR intersection phases, and of course the actual accomplishment of the effort not only sets down a foundation of both methodological best practice, it also creates a large foundation of enterprise metadata that can be used, evolved and expanded through other Enterprise IT scenario efforts. If all this metadata existed in the late 1990s, there never would

have been a “Y2K” crisis. To wit, for a world-wide enterprise was divided into two IT organizations. One had a multi-million-dollar “Y2K" cost and the other’s IT “Y2K” cost was zero. The difference was directly attributable the existence of these types of metadata management and methodology efforts.

7.6 Way Ahead for Enterprise Integration Projects

A most critical failing from the multitude of “Y2K" projects was seeing these efforts as simply projects instead of a reasoned beginning to engineer a long range IT management effort. Instead, the “Y2K” crisis was identified, addressed, and the lessons quickly forgotten. Thereafter, the enterprises and their IT organizations just “moved on” the next project and/or crisis de jour. It is now close to 15 years later. Is enterprise IT any better off? Do we have data and process semantics integrated, interoperable, and non redundant? Or at least better than we were 15 years ago? Most organizations, this author surmises would sadly say, no.

A key accomplishment of “Y2K" was the complete inventorying of their IT systems and databases, analyses of these systems, and the development, storage and use of the derived specifications of these IT systems and databases in order to resolve the “Y2K" crisis. However, shortly thereafter, the very solutions that addressed the “Y2K” problem fell out of favor and were forgotten. There was no infrastructure put into place to continue enterprise-level metadata management, methodologies were not changed, metrics that support quality estimates were not adjusted, deliverables and time-lines were not modified, and finally, “upper” and “lower” CASE software were not put into production. A quick review of Figure 8's metadata domain shows that a majority of these work products are very durable and long lasting.

Sadly, because all this was not done across all of enterprise IT, productivity and quality were reduced, and costs and risks were raised. As evidence of these conclusions, the Chaos studies done yearly by the Standish Corporation have not improved significantly.

But hope springs eternal. The approach posited by this paper can contribute to the development of a significant quantity of high-quality enterprise-level metadata across the entire Knowledge Worker Framework.

7.7 Projects versus Continuous Flow

Once the march to the HR ERP project comes to a conclusion, the never ending cycle of evolution and maintenance starts. When it starts, there is a fundamental shift in the nature of the effort. That is, from project-centric to release-centric. This change is illustrated in Figure 16. Notice that the Metabase System is at the center of this effort. Surrounding the Metabase System are four work efforts.

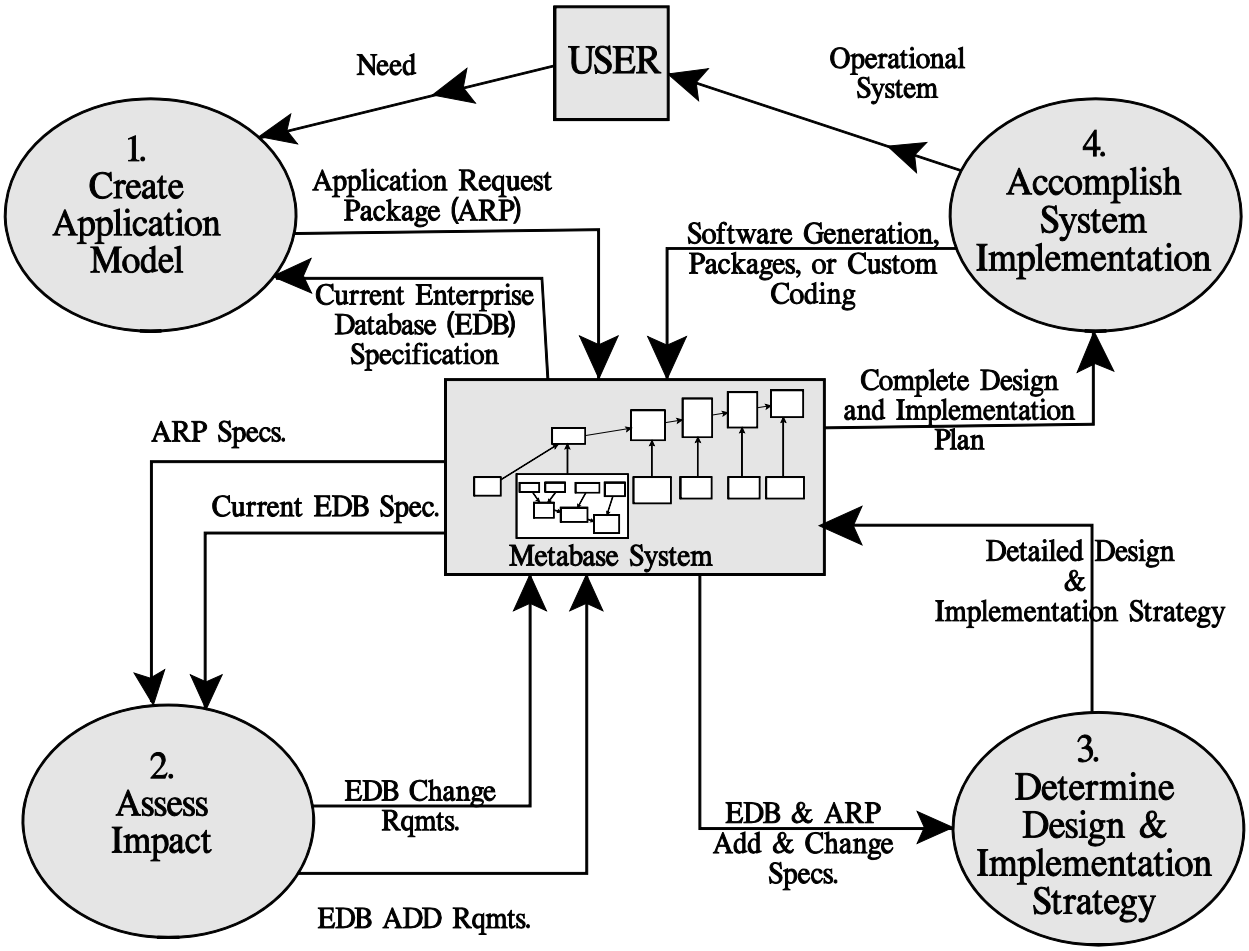

Figure 16. Release centric development.

Create Application Model. This causes the identification and specification of changes needed to one or more aspects of operational business information systems and databases. Note that these application models depend on the Metabase System for a clear understanding of the "what is" and "what should be."

Assess Impact. This causes a periodic assessment of the desired application model changes and the determination of what really needs to be changed. It is likely that more than just the application model needs to be changed to accomplish a collection of application model change requests. These too are stored in the Metabase System's work products.

Determine Design and Implementation Strategy. This takes a determined change-specification and determines exactly what has to be changed and how long it takes. These become real project plans that need funding and approval.

Accomplish System Implementation. This represents the activities undertaken to implement and release changes to the enterprise set of business information systems and databases.

From these descriptions, it should be clear that the entire concept of a business function based project has disappeared and has been replaced with the periodic release of collections of changes to a multitude of business information systems and databases across a multitude of business functions.

Note also that the Metabase System and database is a the very center of this effort. That is because it represents the complete set of enterprise IT intellectual property models in an integrated, interoperable and non redundant manner.

8.0 Approach Q.E.D.

A key question that readers must be asking is whether this paper's approach has been tried and what were the results. Here are some examples:

Industry	Focus
CareFirst (Blue Cross and Blue Shield)	Employed partially developed, but traditionally created specifications to build a metabase-stored data model, which was then used to generate an operational prototype of the needed business information system. A series of demonstrations enabled refinements and regenerations and the ultimate creation of a thoroughly detailed specification that only needed to be implemented once.
Food Services Industry, Hershey	Designed and implemented a Metabase System employing Hershey's DBMS. Used this Metabase system to create, interrelate, and report Hershey's mission, organization, function, data architecture, and resource life cycle models. These models were then used to develop an overall Hershey Information Systems Plan and thereafter several data and process models.
Independent Agency, Federal Home Loan Bank Board (Freddie Mac)	Developed an overall design for a Freddie-Mac wide Metabase System and Database that employed their existing DBMS. The system, once implemented was to store all enterprise specifications for all their business information systems.

Industry	Focus
Independent Not For Profit, MITRE Corporation	Developed and implemented significant enhancements to the Oracle Designer CASE tool to so that the MITRE data models could be created. Created as well were the mission, organization, function, data architecture, and resource life cycle models, which together enabled the development of an overall MITRE enterprise data model and information systems plan.
State Government, California	Developed a Metabase System and database using Information Builder's PC Focus. California had a state-wide management information system to serve the needs of the developmentally disadvantaged. The contractor had completed the system but there was zero end-user or system documentation. The created Metabase System was employed to reverse engineer the entire business information system and use the PC Focus' reporting capabilities to produce complete documentation. Thereafter the Metabase System database was employed as the foundation on which to posit a large scale functional enhancement.
State Government, Delaware	The state needed assistance in the creation of an overall document that contained the mission, organization, function, data architecture models, and resource life cycle models in support of the development of a specification that was to be let for bid by contractors. A version of the Metabase System was developed using IBI's PC Focus.
U.S. Government, Commerce Department	The Commerce Department's Economic Development Administration needed to create a modernized system for its use to manage grants and loans to States. A Metabase System and database was developed using IBI's PC Focus. It stored the work products of the staff assigned to the project. These work products includes mission, organization, function, data architecture, and business information system models. The ultimate work product were architecture and design reports that were delivered to the Government for review and acceptance. Changes were quick and easy to accomplish. The resultant system was implemented by another contractor.
U.S. Government, DoD Office of Inspector General for Audits	The government organization had issued a business information system implementation contract but after about a year, the developed system was declared to be a failure. A second and third attempt was undertaken. The result, failure, was the common outcome. The head of the agency was challenged to undertake the development of a thoroughly prototyped version of the effort prior to the creation of the business information system by another contractor. Developed were the Mission, Organization, Function, Data Architecture, and Business Event, and Business Information System work products from the first three rows of the Knowledge Worker Framework. Created too were a complete set of operational prototypes. Thereafter an RFP was created, awarded, and was successful on the first implementation.

Industry	Focus
U.S. Government, Justice Department (U. S. Marshals Service)	An existing operational system that was critical to this organization was absent any database documentation. No specifications or definitions existed for the current and past two versions of the two databases that comprised the overall data architecture. The Metabase System was employed to import the Operational Data Model schemas that were generated into SQL DDL from the Oracle databases. These Operational Data Model schemas were promoted first to Implemented Data Models, and then through a specially written process, the unique data elements across the tables were uncovered. A three-version old data dictionary was uncovered that had been implemented in Microsoft Access. The columns from the various tables contained basic definitions. These definitions were used to populate the definitions of the Metabase System database's contained data elements and were then, through forward engineering, used to define all the commonly name columns across the two Operational Databases. 80%+ of all the columns then had definitions.
U.S. Government, Social Security Administration	A contractor was charged with developing a complete set of specifications for the implementation of two different business information systems. The effort was to be accomplished through a process-centric data flow diagram methodology. The overall effort was attempted twice. Each agency-review resulted in a failing grade for the contractor. Whitemarsh was requested to determine what was being done wrong. The problem simply was that the methodology needed to be data-centric, not process-centric. A new Metabase System design was created within the existing CASE tool environment, and the requirements analysis, specification, and design activities re-accomplished through the existing staff. The result was delivered to the Government and was judged successful. The follow-on implementation contract was awarded to the contractor. Another key benefit from this effort was the reaffirmation that data-centric is inherently more efficient than process-centric. The quantity of work products for the failed process-centric specifications was compared to the quantity of work products for the data-centric specifications. The quantity were 4.6 times fewer for the data-centric than for the process-centric.
U.S. Government, U.S. Army	A contractor to the army deployed the Metabase System to bring about a consolidated generic database and business information system specification from which weapon system specific logistics systems were created. Instead of each system costing $400K (1985), the first cost $320K, the second cost $160K, and thereafter each cost $80K Savings were directly attributable to the Metabase System and the standardization of the business information system development methodology.

Table 9. Examples of Approach Applicability

9.0 Summary and Conclusions

In the early 1990s, the author of this paper was working for The MITRE Corporation in the Washington, D.C. area. The assigned task was to create an enterprise model for MITRE. That model consisted of Mission, Organization, Database Object Classes, and third normal form models for all the critical MITRE infrastructure data. Once this was completed, the task was expanded to create a MITRE Information Systems Plan that would guide the development of all infrastructure business information system development. The staff for this MITRE infrastructure project consisted of two full time and two part time. The enterprise model and the information system plan models were competed in six calendar months. All this was being accomplished in advance of implementing an enterprise-wide ERP package.

Early on, during functional expert interviews in the MITRE Bedford, Massachusetts offices, the interviewed person asked, "What's the project number?" That was a clear indication that "his" time was to be charged to the project. His justification for charging for his time was this: In the early 1960s, interviewers asked him what his job duties, processes, outcomes, employed data were. These same questions were asked in the 1970s. And in the 1980s. Now, here was in the 1990s. He was tired of answering these same questions over and over, and for free. Simply put, he wanted to know when were we every going to write it all down so we would not have to ask these same questions again, and again, and again.

It is to this very end that this paper has real value. A review of the work products in the columns and rows of Table 1, if recorded, are extraordinarily valuable. They represent the enterprise's IT intellectual property.

For example, with respect to the HR ERP migration scenario, the three sets of work products (see Table 7, Phases 1, 2, and 3), one set for legacy, another for the enterprise model, and a third for the ERP HR system are not only valuable for the purposes for migration from legacy to ERP (Phases 4, and 5), they enable the enterprise to know itself.

They are its missions, organizations, and functions. Its database objects, business information systems, and all the business events that represent the interfaces between the ideal and apolitical missions, database object and business information systems and the tactical and operational business organizations and functions.

The only questions are 1) whether these work products are to be integrated, interoperable, and not redundant, and 2) whether they are stored in a Metabase System an database so that they can be evolved over time and can be use as a foundation for increasing productivity and quality while, at the same time, reducing risk and cost.

Properly intersected and maintained through time, these enterprise metadata models have definitely brought about increases in productivity and quality coupled with decreases in cost and risk many times more than their costs.

Properly accomplished and properly supported through upper and lower CASE environments, these enterprise-level metadata system and databases efforts represent close to a "silver bullet" in real, auditable savings.

Everything described in this paper is not only possible, it has been done as evidenced through the Section 8, Q.E.D descriptions. What remains now is whether you, the reader will cause it to happen to your enterprise. Will you?

10.0 References

The following references to Whitemarsh materials provide a more detailed exposition practical application of the significant content of this paper.

The following documents are available free from the Whitemarsh website:

Topic / Title	URL
Business Event Management	http://www.wiscorp.com/sp/sp16.pdf
Challenge for Business Information System Success	http://wiscorp.com/sp/sp19.pdf
Comprehensive Metadata Management	http://www.wiscorp.com/ComprehensiveMetadataManagement.pdf
DAMA 2002 - Metadata Architecture for Enterprise Wide Data Sharing - Problem Specification	http://www.wiscorp.com/DatabaseDesignInformation.html
DAMA 2003 - Metadata Architecture for Enterprise Wide Data Sharing - Problem Solution	http://www.wiscorp.com/DatabaseDesignInformation.html
Data Management Conference	http://www.wiscorp.com/wrad2000.zip
Data Modeler Architecture and Concept Of Operations	http://www.wiscorp.com/MetabaseDataModelerArchitectueandConceptofOperations.zip
Database Architecture Classes: sample	http://www.wiscorp.com/DatabaseDesign.htm
Database Project Work Breakdown Structure – sample	http://www.wiscorp.com/DatabaseProjects.htm
Earned Value Management	http://wiscorp.com/sp/sp15.pdf
Engineering and Managing Information Systems Plans	http://wiscorp.com/sp/sp11.pdf

Topic / Title	URL
Enterprise Architectures	http://wiscorp.com/sp/sp08.pdf
Information Systems Planning: Book, Course, and Presentation (short and long) – samples	http://www.wiscorp.com/metabase_demo.html
Iterations of Database Design	http://www.wiscorp.com/iterations_of_database_design.pdf
Knowledge Worker Framework: Book, Course, and Presentation (short and long) – samples	http://www.wiscorp.com/metabase_demo.html
Managing Data Models	http://wiscorp.com/sp/sp04.pdf
Manufacturing Project Plans	http://wiscorp.com/sp/sp12.pdf
Metabase Overview	http://www.wiscorp.com/metabase.zip
Metabase System (Free Version) Request form	http://www.wiscorp.com/freemb.html
Metabase User Guides	http://www.wiscorp.com/MetabaseUserGuides.zip
Reference Data Management	http://wiscorp.com/sp/sp17.pdf
Resource Life Cycle Analysis: Paper	http://www.wiscorp.com/MetabaseProducts.htm
Resource Life Cycle Analysis Metabase Module User Guide	http://www.wiscorp.com/metabase_demo.html
Reverse and Forward Engineering Guide Metabase Module User Guides	http://www.wiscorp.com/metabase_demo.html
RFP Development	http://www.wiscorp.com/sp/sp18.pdf