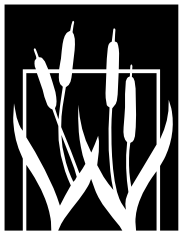

Whitemarsh
Information Systems Corporation

Knowledge Worker Framework and its Deployment through the Metabase System

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

1. The Knowledge Worker

A knowledge worker is someone who primarily works with information and abstract concepts. Another type of worker is the real product worker. White collar workers such as clinicians and clinical support personnel are knowledge workers because they develop care plans, provide treatments, and record results. Administrative staff is also a class of knowledge workers that includes executives, administrators, data processing/information systems personnel, and most other office workers. Alternatively, workers on a manufacturing line and for example, food service personnel are not knowledge workers because they are primarily focused on the creation and/or assembly of real products.

Both knowledge workers and real product workers share common characteristics including plans, schedules, estimates and result assessments. Notwithstanding, the fundamental work methods and environment that underlies the knowledge worker and the real product worker are different at the core. Thus, trying to make one a clone of the other is both frustrating and invalid.

Due to the abstract nature of their work, information required by knowledge workers can best be stored, assimilated and used as objects, which are encapsulations of data, processes and business rules. To most effectively support knowledge workers, the enterprise should strive to create object oriented environments.

These two concepts, knowledge worker and object oriented environments are brought together into technology architectures since both uniquely characterize the ideal working environment.

The knowledge worker's environment involves both automated and non-automated activities. Some non-automated activities involve the use of automation, for example, once a patient receives a treatment from a clinician (non-automated activity), the characteristics of the treatment, and the clinicians observations about the patient's reaction to the treatment are typically recorded in some automated system. A knowledge worker's framework must therefore address manual and automated activities.

Knowledge workers perform groups of functions to accomplish their designated job or to accomplish some aspect of the enterprise's mission. Knowledge workers may perform these function groups in different combinations depending on the enterprise's organization. For example, if an organization is highly distributed into multi-functional units, there may be staff that perform diverse groups of functions. Conversely, a highly centralized organization may have certain staff devoted to specific and highly specialized functions. The knowledge worker is therefore a complex multi-faceted person who performs diverse functions of different complexities for one or more organizations.

Enterprises commonly create computing supports for knowledge workers under the assumption that the functions they perform and the organizations through which they act are fixed and seldom change. Not only are these assumptions wrong, but when the functions and organizations do change, computing environment changes seldom keep pace because they are time consuming to specify, difficult to implement, and slow to accomplish. Slow-to-react

computing environment changes, therefore, become the very reason why information technology support to business functions and organizations cannot keep pace with the demands of change. What is needed are computing environments that are object oriented, sensitive to knowledge worker functions and organizations, and that can react to the demands of change in a timely fashion.

2.0 A Framework for the Knowledge Worker

The Knowledge Worker Framework was constructed inductively from the Whitemarsh Database-centric Application Methodology. The methodology consists of six phases, which are:

Methodology Phases	Key Work Products
Preliminary Analysis	Missions, Organizations, Functions, Hi-level Data Model, Information Systems Plan.
Conceptual Specification	Detailed Data and Process Models, Interrogation requirements, and System Control requirements. Several cycles of complete prototyping resulting in design changes.
Binding	Choice of DBMS engine, Design changes caused by engine,
Implementation	Complete implementation through to unit, system and integration testing.
Conversion and Deployment	Migration of existing data along with training and documentation.
Production and Administration	Production system startup and cycles of evolution and maintenance along with comprehensive auditing.

Business information system work products are organized into different groupings related to the overall mission, data-centric, business information systems centric, functions and organizations. Added to that is the intersections between business information systems and the human functions that are performing knowledge worker based efforts.

Development of the work products is accomplished through interviews and involvement of different sets of “users,” including executive management, managers, end user, power users, systems analysts, programmers and data-specialists.

In addition to different collections of users creating different work products, the general notion of the business information system is unfolded through different perspectives which are: scope, business, system, technology, deployment and operations.

When all this was brought together, a framework for the knowledge worker emerged. It, the Whitemarsh Knowledge Worker Framework (KWF), is depicted in Table 1. Shown is a perspective column, and then six specific columns for Mission, Database Objects, Business Information Systems, Business Events, Business Functions, and Business Organizations.

Its exposition, while necessary is however not sufficient. The framework must be valuated as being the necessary framework in terms of rows, columns, and cells of work products to know that if followed will result in correctly designed and operational business information systems.

The process of validation was conducted through the identification of examination and audit reports of over 10 multi-$100 million dollar U.S. Government IT system failures. These reports are located on the website of the U.S. Government Accountability Office. There were over 125 unique failure findings across these audit reports. Failure findings that were similar were not counted multiple times.

The KWF framework was used as the allocation and validation mechanism. That is, GAO IT failure findings were allocated to specific KWF cells, and the work products associated with those cells were examined to determine whether, if properly accomplished, would cause the failure finding be resolved.

When the failure finding was resolved the work product's specifications remained unchanged. When the failure finding was unresolved, the scope, specification, and work steps association with the failure finding were modified until the failure was resolved.

There was a second outcome from this allocation and validation effort. It was the determination of the overall percent of failure-findings shown in each of the KWF cells. These percent amounts are shown in Table 2.

There are significant relationships across the columns of the KWF. These relationships are shown in Figure 1. The six columns are across the bottom of Figure 1. The meaning of the relationship is shown as a rectangle above each pair of columns.

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 and hierarchies	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, responsibilities, 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							
	Mission	**Database Object**	**Business Information System**	**Business Event**	**Business Function**	**Organ-ization**	**Row Totals**
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. Totals	**14**	**7**	**7**	**3**	**37**	**32**	**100**

Table 2. Percent representations of GAO Identified Reasons for IT System Failure.

The key take away from Table 2 is that 41% of all IT systems failure during the first two rows of accomplished work products. That is, during the Scope and Business rows. 50% of all IT system failures occur after the IT system begins production. Those are the failure percent shown in the

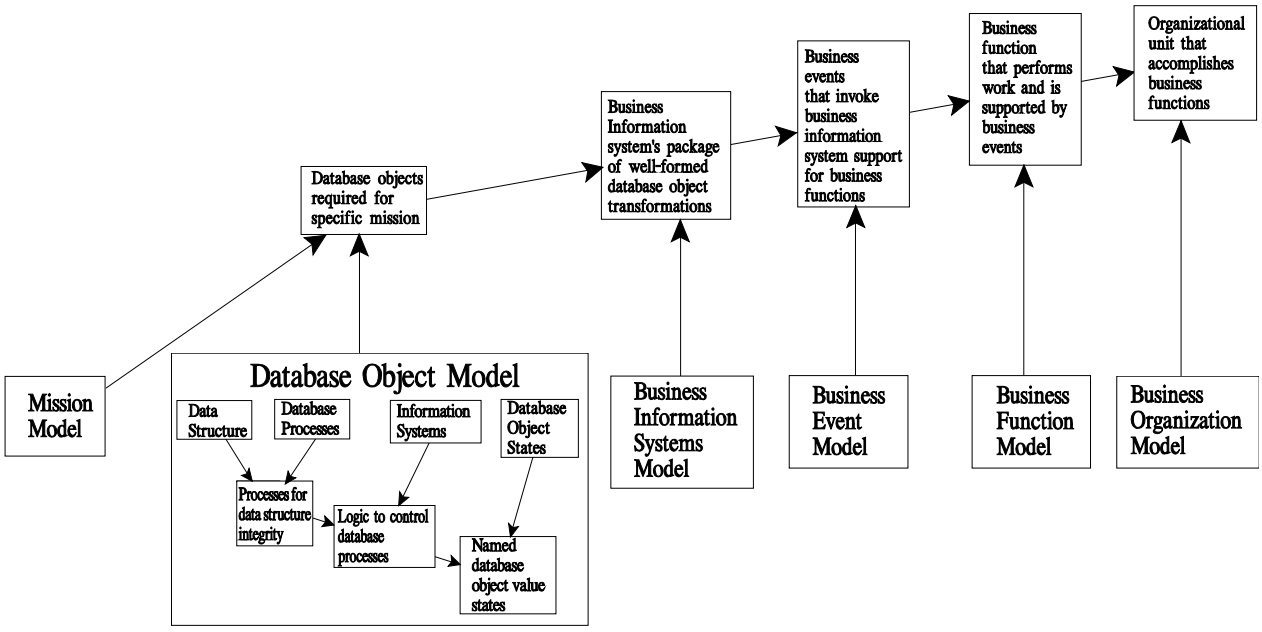

Figure 1. Relationships across the columns of the Knowledge Worker Framework.

Organization and Business Function columns for the System through Operations rows. What is left are the reasons IT is actually responsible for the IT system failure. 8%.

3.0 Description of The Knowledge Worker Framework

3.1 Perspective Column

The perspective column provides context for the product columns's cells. The six rows of this column are scope, business, system, technology, deployment, and operations. Each of these six rows classifies the vantage point from which the work products cited in the Knowledge Worker Framework cells are viewed.

Each of the six perspective rows represent an unfolding of the work products related to their overall parent. For example, there is a Mission scope, then hierarchies, then policy hierarchies, policy execution enforcement, installed business policy and procedures, and finally, the operating business.

While it is ideal to specify and implement Knowledge Worker Frameworks in a top-down, left-right fashion, *life* and businesses seldom wait. Even more importantly, knowledge worker environments are already operating, but not optimally. Given that only a few Knowledge Worker Frameworks are done top-down and most are operating in an as-is, possibly chaotic fashion, businesses must re-conform their Knowledge Worker Frameworks to an ideal, top-down fashion to achieve maximum benefits.

If an environment is chaotic or less than optimum, analysts must first identify and allocate all the existing knowledge worker products into the framework. Once identified and allocated efforts can begin to transform the framework from its current form to one that would have resulted if originally done in a top-down fashion.

The message is simple: no matter the current state of a businesses knowledge workers products, they must be identified, allocated and optimized. Doing nothing or benign toleration should never be accepted. Effective and successful enterprises are seldom easy. There are no non-fatal encounters with bullets to the head, be they silver or not.

3.2 Mission Column

The mission column represents the rationale or basis for the knowledge worker environment. The first cell, scope presents the list of missions. The set of missions are those that form the basis of the enterprise. If a mission is missing then so too is an important aspect of the business. Missions are either external or internal. External missions are those that support the income of the business. Internal missions are those that employ the business's income to operate the business in support of its external missions. For example, if the external mission of the business is to sell a specific product line, then the internal missions are those that support sales, for

example, human resource management, research and development, manufacturing, inventory and distribution, and sales management.

Missions are mechanisms for enterprise database partitioning. Once missions are listed they become the criteria for including or excluding entries in the remaining 35 cells. Additionally, once missions are delineated, then one or more missions can be chosen to pursue through the remainder of the framework. Each mission may also be pursued by different analysis and design teams. The only real down side to this approach is integration once different subordinate missions implementations are accomplished. If the top two rows (scope and business) are completed prior to breaking the work into separate teams the end result is more easily integrated.

The business view of a mission contains mission hierarchies. Each mission, for example, product sales, or human resource management is represented as a hierarchies of text paragraphs and is presented in an "accomplished-form." That is, the mission is described as if it were completed in an completely ideal manner. Completely removed are any indications of either WHO or HOW.

The systems view contains hierarchies of the policies that must be present to accomplish various missions. Business policies that must present to accomplish an enterprise's mission.

The technology view represents fully specified and implemented view of enterprise policies that are executed and/or enforced.

The deployment view represents the actual "in the field" sets of policies whose execution result in data that is collected, updated, and reported. Data is executed policy.

The final row, operations represents the ongoing and executing set of policies that carry out various aspects of the enterprise' missions.

3.3 Database Object

A database object is a collection of traditional (that is, formatted and structured data) and nontraditional (that is, video, sound, and unstructured text) data presented to requesters. Database objects proceed through precisely defined states starting with the null state, and then a series of discrete business defined, interlinked non-null states, and finally a null state. Database objects are squarely based on policy analysis for its data structure formulation, and on procedure specification for the proper valuation, modification, migration, and reporting of database objects.

The scope view of database objects starts with listing the different business resources[1] employed in the enterprise. A resource is a fundamental of the business about which information

[1] A business resource, according to Ron Ross in *Resource Life Cycle Analysis, A business Modeling Technique for IS Planning* (Database Research Group, Inc., Boston, MA 1992) is "something of significant scope, substantial complexity, and enduring value that has actual substance and/or structural design, which enables the business to achieve its mission and carry out its functions. These functions manipulate, exploit, direct, and/or transform such resources.

is collected, funds are expended, or is sold and expensed. Examples are people, contacts, fixed assets, and the like. All business resources are set squarely within the business' missions.

In the business row, the business resources are decomposed into their business resource life cycles. Resource Life Cycle (RLC) was developed by from Ron Ross.[2] Resource life cycles form the basis of the information system plans. Each resource life cycle contains the major state names from the business resource's life cycle. From Ron Ross' book, the resource life cycle for parts might be:

- Define part types
- Establish suppliers
- Acquire parts
- Accept part requests
- Ship parts
- Maintain parts

Resource life cycle examples are provided in Chapters 5, 6, and 7 of the Whitemarsh *On* Database book, *Database Objects, The Foundation Stones of Enterprise Database*.

The system's row contain the fully specified database objects. A database object consists of four parts: data structure, database object process, database object information system and database object state.

- **Database Object Data Structure**: the set of data structures[3] that map onto the different value sets for real world database objects such as an auto accident[4], vehicle and emergency medicine incident. Each data structure contains fields, data integrity constraints, and if tables, columns, table, and other types of constraints.

---

2 *Resource Life Cycle Analysis, A Business Modeling Technique for IS Planning* (Database Research Group, Boston, 1992) is a technique for identifying the components of a business that is subject to information systems. The resource life cycles are the basis from which database objects are identified, designed, implemented and deployed.

3 A database object data structure has an identifier to isolate its instances from all others and fields which represent single values, multiple values (i.e., vectors), groups, repeating groups, and nested repeating groups. When a data structure only contains single valued fields it is termed a simple data structure database object. If the data structure contains multi-valued fields though nested repeating groups then it is termed a complex data structure database object.

4 One of the standard Whitemarsh examples for database objects is based on a state public safety agency that was establishing a state-wide system that embraced a multi-tiered set of client-server systems. Each agency (e.g. state and local police, roads, licencing, inspections, and emergency medicine) had their own client-server systems. The state maintained the large scale agency-state client-server system that coalesced data, regulated valid values, created state wide statistics, etc.

- **Database Object Process**: the set of database object processes that enforce the integrity of data structure fields, references between database objects and actions among contained data structure segments, the proper computer-based rules governing data structure segment insertion, modification, and deletion. For example, the proper and complete storage of an auto accident.

- **Database Object Information System**: the set of specifications that control, sequence, and iterate the execution of various database object processes that cause changes in database object states to achieve specific value-based states in conformance to the requirements of business policies. For example, the reception and database posting of data from business information system activities (screens, data edits, storage, interim reports, etc.) that accomplish entry of the auto accident information.

- **Database Object State**: The value states of a database object that represent the after-state of the successful accomplishment of one or more recognizable business events. Examples of business events are auto accident initiation, involved vehicle entry, involved person entry, and auto accident DUI involvement. Database object state changes are initiated through named business events that are contained in business functions. The business function, auto accident investigation includes the business event, auto-accident-incident initiation, which in turn causes the incident initiation database object information system to execute, which in turn causes several database object processes to cause the auto accident incident to be materialized in the database.

The technology view represents database objects completely through SQL. As SQL moves from SQL/92 to SQL/3[5], less and less of the facilities of database objects will be proprietary. Today, the majority of the database object's database object processes and database object information systems are SQL vendor proprietary. Notwithstanding the quantity of vendor proprietary code, it is all commercial off the shelf (COTS) software, and firmly based on the technology independent database object specifications contained in the systems view.

The deployment view for database objects are the actual instances of distributed SQL based information systems. The final row, operations view, represent the running database object based information systems.

[5] Despite the standardization of SQL/3, end-users will be unable to determine whether products actually conform to the standard because the National Institute of Standards and Technology (NIST) has abandoned its public law mandated role of standardization. End-user organizations will have to perform their own conformance testing activities which could cost upwards of $600 thousand per year per end-user organization. For more information on this sad event, see the Whitemarsh web site.

3.4 Business Information System Column

A business information system is a computer based data processing system that accomplishes database object state transformations from within the context of business functions. Different business functions may cause the execution of the same business information system. If, in any of the business functions that employ a business information system, the database object value state transformation is not accomplished, the entire set of database object transformations are rolled back so that the database object returns to its prior state.

The scope view identifies the business information systems required to support the business resources (cell to the left). The list is simple, one business information system per business resource[6]. If a business resource is people, then the business information system would be the human resource management information system. Similarly identified and named are finance information systems, customer management, facilities management, project management, and asset management.

The business view for business information systems contains the business information system hierarchies necessary to carry out the information system requirements of the database object transformations inferred by the business resource life cycles. Named components within each detailed business information system clearly identify the nodes within each resource life cycle. For the parts business resource, the necessary business information systems might be a depicted in the table that follows.

Parts Resource Life Cycle Nodes	Business Information System Hierarchies
Part type definition	Create part type, Maintain parts (insert, maintain, and delete part)
Supplier establishment	Create supplier, Maintain suppliers (insert, maintain, and delete supplier)
Parts acquisition	Enter part receipt Adjust inventory
Order management	Reserve part for order Adjust order line item Report inventory status
Parts shipment	Build bill-of-lading Establish shipment Acknowledge shipment receipt
Parts maintenance	Adjust parts inventory Replace existing inventory

[6] Within the technology row, each business information systems from the system's row may become multiple information systems that are implemented on different hardware, operating systems, and that operate through different DBMSs.

Within the systems view each identified business information system component, for example, Adjust parts inventory, is detailed in terms of its logic, screens, file accesses, and reports. Included at this level are the necessary connections to the specific aspects of each database object data structure. Identified and connected as well are the database object processes, and the necessary database object information systems that begin and structure the processes necessary to modify the database object's structure from null to an allowed non-null state.

The critical difference between the database object information system and the business information system is that the database object information system is completely specified and totally implemented within ANSI standard SQL language while the business information system is completely specified and totally implemented within either an ANSI standard 3GL (e.g., COBOL, C) or a vendor proprietary 4GL (e.g., Clarion, Delphi, FOCUS, and Power Builder).

The reason that the database object information system is expressed entirely in ANSI/SQL syntax is so that it can be ported from one ANSI standard conforming SQL DBMS, operating system, and hardware platform to another. In contrast, the business information system directly interfaces with end users. In client/server parlance, that means that the business information systems is in the "presentation layer."

From the technology view, business information systems consist of traditional components, that is detailed designs for screens, files, reports, processes, menus, and the like. To accelerate business implementation, code generators should be used whenever possible. Assuming they are, four benefits immediately accrue:

- Detailed design is quicker because the code generator builds so many of the system components.
- Coding errors within generated programs are virtually eliminated thus making unit test time close to zero.
- Generated system design documentation is commonly an automatic by-product of code generators
- Long term maintenance is easier because of the three previous benefits.

The business information system components that exist from within the deployment view are also traditional. For example, the actual systems, programs, menus, and data files. When these components are generated, these only have to undergo normal configuration management.

Finally the components that exist within the operations view are the executing systems that take in data, produce reports and perform calculations.

3.5 Business Event Column

Business events are the interface between the two "machine" columns of the framework and the two "man" columns. The main reason there is a column formally dedicated to the interface between man and machine is to preserve their independence. The "man" columns are able to be crafted to fit different and individual functional styles within different and unique organizations. The intent of the Zachman framework is to architect and configure information systems. In contrast, the Knowledge Worker Framework embraces the entire knowledge work environment in which information systems play only a part.

The scope view contains the list of business events that are required to accomplish business information systems as they support business functions. For the parts example, above, the business events are: Perform parts acquisition and maintenance. Each listed business event acts as a surrogate for the set of business event sequences and if necessary, hierarchies. Each sequence or hierarchy is represented by one member of the business event list.

The business view contains the various business event sequences and hierarchies. For example, using the parts example from above, the information in the columns surrounding parts are:

Resource Life Cycle Node	**Business Information System**	**Business Event**	**Business Function**
Part type definition	Create part type	Invoke part creation	Create new part information for business
Supplier establishment	Create supplier,	Invoke supplier creation	Establish new supplier of parts
Parts acquisition	Enter part receipt	Invoke part receipt	Acquire parts from supplier

The systems view of business events are the specifications of the invocation protocols, input and output data, and the various messages that must be exchanged between the business information systems and the business functions. In the example of parts, the input information are the specifications of the data that must be submitted to establish a new part or supplier, or the specification of a report that is produced by a business information system in support of a particular business function.

The technology view for the business event are the precise specifications of the man-machine interface for the different types of involved technology. For example, one business function may cause the creation of the data necessary to instigate a batch report. Another business function may have to create input data in a specific sequence and format. A final example might be the format and the mode of a generated report.

The development view a business event are the actual developed forms, computer screens, data entry instructions, and the instructions for acquiring and handling reports.

The final view, operations, embraces the day to day operational aspects of interfacing business functions and business information systems. This involves ensuring that there are enough data entry forms, sufficient paper for reports, computers, and telecommunications networks.

3.6. Business Function Column

A business function is a procedure accomplished by someone within an organization to complete some aspect of a business' mission. Business functions almost always exceed the bounds of business information systems. For example, a business function to acquire a new part of a company's inventory might involve identification, gathering examples, analysis for engineering, durability, cost, and repair. Finally, a part is selected for inclusion. Then and only then is the information about the part encoded onto a data entry form as required by the appropriate business event, and then entered into a database through a business information system.

Business functions are commonly a matter of style. Different business organizations can have the same business function style, and the same business organization can have different business function styles. The greatest disaster that can befall a large scale information system is that it's design is derived from a hierarchical decomposition of the business function's lowest levels. When that happens and there is the slightest change to the business functions, the business information systems must also change. The business information system get's whip-sawed. Or, stated differently, whenever the business functions get a *cold*, the business information system, at best gets a *pneumonia*, and at worst, *dies*.

The Whitemarsh Knowledge Worker Framework is engineered to keep business information systems and business functions independent one from the other. Only when the business functions change to the extent that they need additional or different business information systems are business information systems impacted by business function changes. These changes typically occur only after there has been a business mission change.

The scope view of business functions is the list of the highest level business functions. These functions should closely parallel missions. Missions however, are different from business functions. Business missions are the ultimate target of the enterprise. Not all missions are necessarily accomplished in the manner they are described. Business functions, however, are always accomplished else the business does not operate.

Business functions change over time far more frequently than do business missions. Consider for example any large insurance company. Clearly their missions deal with finding clients, offering insurance, performing underwriting, selling and administering policies, and paying claims. These missions have been the basis of insurance for several hundred years.

Business functions however, may change far more often. Insurance almost certainly was only sold through direct contacts with insurance agents. Today solicitations come in the mail almost every day and the agents call only during dinner. Payments formerly made through the agent who came to the door on a "debit" route can now be automatically deducted from checking

accounts. Claimants used to await the individual insurance agent to inspect damage can now have their claims filed and adjudicated over the phone. Finally, formerly payments presented by the insurance agent can now be wire transferred to claimant accounts.

Sometimes however, the same business function is performed differently by different organizations. Another difference between mission and function is that missions are described independently from the *how* and *who* accomplish them. Business function is the *how* description. Within any business function, the missions are presumed, but the *who* is not known whenever business functions are performed by all business organizations. Whenever the same business function is performed by multiple organizations, but differently, then the business function description can be described in terms of the specific organization.

Each business function, within the business view is described in terms of the scenarios performed to accomplish some aspect of the business' mission. Each business function hierarchy is set down along with the sequencing of the steps within each hierarchy. If different business organizations perform the business function then the scenario descriptions can be different so long as the ultimate objectives of the function are clearly identified and are obvious to those who perform the function.

The system view of business function contains the exposition of the best practices, quality measure, and accomplishment assessments. These materials represent the idealized methods an organization can employ to accomplish business functions. Supporting each best practice are the various performance targets and assessments that judge satisfactory accomplishment. Whenever business functions are performed differently there must be style independent assessments.

From within the technology view, business functions are detailed into their specific activity sequences that accomplish the business scenarios. Each set of activities are stylized to fit the specific organization carrying them out. The activity sequences are evaluated against the best practices and assessment criteria to ensure that the activities accomplish the desired result.

The deployment viewpoint represents the actual office procedures employed by organizations performing business functions. These deployed activities must be supported by necessary operational policies and procedures and what ever technology supports that many be required.

The operations viewpoint are the detailed instructions that exist within an office and a schedule to actually perform the business function's work. These office procedures should be taught, monitored, and constantly evaluated for maximum efficiency, effectiveness, and minimum cost and risk.

3.7. Organization Column

An organization is a formally constituted group of persons chartered to perform business functions to achieve some aspect of a business' mission. While small businesses often have the same organization from one location to another, large businesses do not. In fact, as businesses

become larger and more diversified, organizations become different, stylized, and whenever management changes, business organizations often change in lock-step. Even when the mission of the enterprise fundamentally stays the same, there are business organizations. It is also quite common to change business function to match the new styles adopted by the business organization changes.

Business organizations are capable of change at a far greater rate than can the business information systems that support them. Thus, while organization changes might only cause mild disruptions, business information systems changes occur only after great expense and significant disruption. Because of these two dynamics, it's ideal not to have to change a business information system whenever a business organization and/or business function changes.

To achieve this ideal, the Knowledge Worker Framework is engineered to keep business functions and business organizations independent one from the other, and both independent from business information systems. Only when the business organizations change to the extent that they need additional or different business functions are business functions impacted. The most common changes are those that cause business functions to be either transferred from or into different business organizations. Those changes seldom ever impact business information systems. The only business organization changes that impact business information systems are those that typically occur after there has been a business mission change.

At the scope level, the list of the business organizations performing the business functions is provided.

At the business view level the various organization charts, jobs and their descriptions is provided. These provide an understanding of the types of persons who will be performing the business functions.

Within the system view, the detailed job roles, responsibilities and activity schedules are provided to better understand when and how the business functions are accomplished.

From the technology viewpoint, the various procedure manuals are created along with their task lists, the quality measures that ensure that the activity is successfully accomplished by the specific organizational unit, and the specifications of exactly how the activities are assessed. These materials are created and then updated on an as needed basis.

From the deployment viewpoint, the daily schedules, shift and personnel assignments are created. These are integrated with the various business functions. Since organizations can vary there may be different configurations that perform business functions. The measure of equivalency are the best practices, measures, and assessment criteria created as part of the business function system viewpoint.

From the operations viewpoint, organizations are deployed and accomplish the full set of business functions necessary to carry out the business' missions.

4.0 Metabase[7] System, The Implementor for the Knowledge Worker Framework

No one would ever question why a business needs it's finance books. Well, the metadata repository is the business's information systems' books. If you cannot run a good business without the former, you cannot run good information systems environment without the latter.

A significant portion of the time and costs associated with resolving the Year 2000 problem can be directly attributed to a lack of a quality metadata environment within information systems organizations. The fact that one information system organization within an enterprise had virtually no Year 2000 problem while another organization within that same enterprise was running their information systems shop "24x7" was no accident. The former had a long history of metadata management and the later thought metadata was a wasted overhead expense.

Vital to database success is control over semantics. The controls are mainly in the area of the definitions that form the basis of the interfaces to standard processes (e.g., computing net profit) and the standard data definitions (e.g., what does profit *mean?*).

It is not necessary, however, to control the interfaces to the *end user*. Just how a data entry screen or report looks to different people is immaterial so long as the enforced semantics (rules of meaning and usage) are the same.

In the development of large data processing projects dealing with enterprise-wide, indispensable business functions, documentation of the design requirements and resulting information system specifications is seldom accomplished such that it is timely, accurate, or complete. That is disastrous for the following three reasons:

- Only the momentous facts that are remembered are recorded.
- As systems are specified, the lower-level design details are redundantly developed, often in conflicting manners.
- As system components are maintained, the efforts are crippled because of the undocumented business knowledge that is essential to understanding the component.

Amelioration of these three important problems starts with organizations adopting formal methods for performing analysis and design. Formal methods are only measurably productive

[7]Metabase is a term crafted from "metadata database." The term, metabase, has been used by Whitemarsh since 1981 in reference to the many different metadata database systems that Whitemarsh has built for its clients. These metabases have been built in Information Builder's Focus, CA's IDMS, .Cincom's Total, and SoftVelocity Corporation's Clarion for Windows.

and repeatable if they are very detailed and proceduralized. Such detail, however, dehumanizes knowledge workers, who, in turn, are certain to generate protests about being production workers on an assembly line, which, by the way, is worthwhile only when all of its products are the same. In contrast, to the production line, business information system designs are unique assemblies of large sets of components, many of which are similar in design.

Designing business information systems is not an activity for the production worker; rather, it is an activity for a knowledge worker. While there is clearly procedure to both activities, designing an information system requires individualized applications of creativity, human factors techniques, and rule making. Accordingly, requiring the robot-like use of a fully detailed methodology cannot result in responsive information system designs. Work plans must be drawn from proven techniques against which metrics have been captured and honed over the years.

Building a business information system, once it is designed in sufficient detail, is largely a rote application of computer language coding. There are a number of quality and robust code generators that can use the metadata for a business information system design to produce computer code that is competitive in performance to a human coded application. There is, of course, no comparison between human coding costs and code generator costs.

To fully respond to the three problems cited above, knowledge workers should have the freedom to create their own analysis and design work products for data and processes within strictures dealing with format, time, quality, and resources. These work products must be placed into a metabase. The metabase, containing these products in fixed formats and sequences, can be accessed by code generators (both human and computerized) to build the business information system. If the generator is quick enough, a fully functional version of the business information system design can be live-tested a short time later. As design flaws are found, the metabase's metadata can be changed and the business information system regenerated. *In short, an interactive design process, in which the metabase is the empowering component.*

Traditionally, it is not uncommon to expend 20 percent of a total systems development lifecycle on requirements and design. The remaining 80 percent is expended on building, testing, and documentation. Once implemented, 500 percent more is spent over a system's lifecycle for changes, fixes, and evolutions, also in a 20/80 ratio. The overall total is 600 percent. If, with code generators, the 80 percent is reduced to effectively zero, then there must also be a profound reduction in the 500 percent systems lifecycle maintenance.

4.1 Metabase Functional Modules

The metabase concept implemented as a database application includes:

- Business Information Systems
- Business Events
- Data Elements

- Database Objects
- Documents and Forms
- Implemented Data Models
- Missions, Organizations, and Functions
- Operational Data Models
- Requirements Management
- Resource Life Cycle Analysis
- Specified Data Models
- Use Cases
- View Models

Whitemarsh has implemented these into discrete database applications with Clarion for Windows by the SoftVelocity Corporation (www.softvelocity.com). These metabase systems operate on Windows computing environments. Clarion for Windows was chosen because it meets the Whitemarsh requirements of sophisticated code generators coupled with sophisticated metadata management within its environment. Metabase environments are distributed to Whitemarsh website members in the form of SQL loadable metadata. The data management engine of the metabase is SQL via ODBC. Access to metabases can be through ODBC clients such as Crystal Reports. The SQL engine is can be operated through O/Ss other than Windows.

Metabase System access over the Internet is accomplished through Java-base client software that can operate on any client machine that can run Java "jars." Through this strategy Metabase System users have the same look and feel from the intra-net access as with their inter-net access.

4.2 Metabase Benefits

The complete set of metadata components map onto the complete life cycle of database application, that is, its:

- Specification
- Implementation
- Operation (and maintenance)

The following is a partial list of benefits attained through the use of a metabase. A metabase will:

- Assist top management in identifying the resources required to build an information system.
- Provide discipline and control for the design process.
- Provide a structured approach to conceptual design.

- Enhance the application development process through the utilization of prior work.
- Provide a management facility for monitoring database projects.
- Allow for the non-redundant storage of data definitions and business policies that produce greater consistency throughout the enterprise.

4.3 Relationship between Metabase System Modules and Knowlede Worker columns

Table 3 sets out descriptions of the Metabase System Modules and then via check indicates that one or more work products within the Knowledge Worker columns are created, updated, or employed.

Metabase Software Module		Knowledge Worker Framework					
		Mission	**Database Objects**	**Business Information**	**Business Event**	**Business Function**	**Business Organization**
Mission, Organization, Function Position Assignment		✔			✔	✔	✔
Resource Life Cycles			✔	✔			
Document & Form, Information Needs Analysis, Requirements, Use Cases, and Data Integrity Rule Specification & Binding				✔		✔	
Data Modeler	Data Elements						
	Specified Data Model						
	Implemented Data Model		✔				
	Operational Data Model		✔	✔			
	View Data Model		✔	✔	✔		

Business Information Systems, Reports, and Wire Frames			✔			
Information Systems Planning	✔	✔	✔	✔	✔	✔

Table 3. Relationship between Metabas System Moduels and Knowledge Worker columns

4.4 Business Questions Addressed by the Metabase System Modules within the context of the Knowledge Worker Framework columns

Table 4 over the next several pages presents the key business questions that are answered/accomplished through the use of a given Metabase System module and the Knowledge Worker Framework column(s) work products that are affected.

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Business Event Management: *What are the business events, where are they, how are they related to both the overall business's process model and calendar models, and then now are these events related to 1) mission, organization, function, 2)business information systems, 3) business event cycles, and 4) calendar cycles. What is the impact on these business events when policy (a.k.a., data) is required or changed.*				✔		

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Business Information Systems: *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.*			✔			
Data Element Model: *What are the context independent business facts and their specifications that can be deployed to fully define the semantics that define these business facts, or that form the basis for these business facts as they are refined through the allocation of semantic and data use modifiers. Where are these data element semantics deployed throughout the various data models whose facts (attributes, columns, DBMS columns, and View columns are based on these facts.*		✔				
Data Integrity Rule Specification and Binding: *What are the rules that govern the integrity of data that is specified across all the deployed uses of that data. How are these rules defined and where are they bound such they are defined once and bound where appropriate? What are the processes and how are these processes executed that ensure data integrity during all data object operations?*		✔	✔			

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Database Objects: *What are the major data-based object classes and objects that form enterprise databases. What are the state-based process life cycles for these data-based object classes? What are the database record (i.e., table-based) processes that control the fundamental integrity of individual database records. What are the data-based object class business information systems that transform database objects from one predefined value-state to another?*		✔	✔			
Document and Form: *What documents and forms provide critical information about the enterprise? How are those documents and forms interrelated one with the other? How are these materials subdivided and then properly related to specific functions performed by organizations in the accomplishment of missions? How are these able to be related to certain View columns?*	✔	✔	✔	✔	✔	✔

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Enterprise Architecture Management: *What are the five distinct architectures that comprise the over set within the enterprise, and how are these architectures intersected and mapped to the Knowledge Worker Framework in support of projects that contribute to the build rationale and sequence of an overall Information Systems Plan?*	✔	✔	✔	✔	✔	✔
Functions: *What are the human-based processes performed by groups in their achievement the various missions of the enterprise from within different enterprise organizations? What human processes are common across and within organizations, business events and by indirection, business information systems and databases?*					✔	

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Implemented Data Model: *What are the database models that are able to be employed as the basis of operational databases that can be employed by business information systems? What are the database implementations of data-based concepts contained in the specified data models? What are the various columns, including value domains, data integrity processes, and data types that are an implementation of various data elements and/or attributes from specified data models that defined concepts?*		✔				
Information Needs Analysis: *What information (a.k.a. query results or reports) is needed by various organizations in their functional accomplishment of missions and what databases and information systems provide this information?*		✔	✔			
Missions: *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?*	✔				✔	✔

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Operational Data Model: *What are the actual databases and models that are employed by business information systems? What are the databases and models that operationally specify data-based specifications from DBMS independent database models? What are the various DBMS columns, including value domains, data integrity processes, and data types that are an implementation of various data elements and/or attributes from specified data models that defined concepts?*						
Organizations: *Which organizations are accomplishing what aspects of missions with what databases, information systems and through which functions*?	✔			✔	✔	✔
Project Management. *What are the various knowledge worker projects that address one or more collections of activities that create an IT work product that supports the enterprise? What are the detailed deliverables, collections of tasks, assigned staff, work accomplishment resources, and work environment factors that affect the accomplishment of project work. What are all the different projects by IT work product.*	✔	✔	✔	✔	✔	✔

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Reports. *What are the specifications of the reports that are to be generated by business information systems and/or DBMS ad hoc query language or package report writers. What are the various view columns or directly connected database columns involved in the database?*			✔	✔		
Requirements: *What are the requirements that in total support the development of key enterprise database components? How these requirements are interrelated, subdivided, and then related to the various metadata components that are "required" as a consequence? How can the complete set of effects can be known and interrelated?*	✔	✔	✔	✔	✔	✔
Resource Life Cycles: *What are the key Resources (facilities, materiel, staff, etc.)?How are they sequenced, interrelated, and how are they supported through databases and information systems?*		✔	✔			

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Specified Model: *What are the data model specifications that represent concepts that are to be included in database models? What are the data elements that are represented as subject-entity-attributes across the specified concept data models? What are the various attribute-bound value domains, data integrity processes, and data types that are an implementation of various data elements?*		✔				
Use Cases: *What are the detailed business process scenarios required to accomplish the necessary work of the enterprise? What are the interrelationships among use cases? How are the use cases subdivided into certain events? What are the pre-, post-, and special-conditions of these use cases? What are the business facts that are read, selected, updated, and reported within use cases? What are the relationships between use case facts and database view columns?*	✔	✔	✔	✔	✔	✔

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
User Acceptance Tests: *What are the requirements-based accepted tests deemed appropriate and necessary to determine the adequacy of a business information system before it is released? What are the requirements that are being validated as being completed? What are the business information system components that are being tested?*		✔	✔			
View Model: *What are the data interface specifications between databases and business information systems? What are the various business information systems that are supported by specific databases? What are the various databases that are accessed by business information systems. What are the data-based mappings between views that support database interoperability? What are the data-based processes that transform data via one view to the data specifications of a different view?*		✔	✔			

Business Questions Addressed by the Metabase Modules Within the Knowledge Worker Framework						
	Knowledge Worker Framework Columns					
Metabase Software Module Scope (alphabetically listed)	Mission	Database Objects	Business Information Systems	Business Event	Business Function	Business Organization
Wire Frames: *What are the end-user graphical user interface specifications that enable end-users to interact with business information systems for data entry, formatted reports, ad hoc reporting, and for navigating through defined features of the business information systems? What are the various screens that are affected by specific view-column database interfaces with business information systems? What are the various processes that affect either the data presented through the wire frames or are stored in the databases? What are the various controls and invoked processes evident through the wire frames?*		✔	✔			

Table 4. Business Questions Addressed by the Metabase System Modules within the context of the Knowledge Worker Framework columns