Whitemarsh
Information Systems Corporation

Project Metrics

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. Introduction and Objectives

The objective of this paper is to present Whitemarsh's approach to the construction of project metrics.

Every work-product associated with a database project takes a finite amount of work time. After many years of observations, the quantity of staff hours has come to be predictable when the methodology is standardized, the work-products are identified and engineered, when the work environment is normal, the staff and skills are adequate, the equipment is normal, etc. The work effort associated with each work-product is called its Work Unit Effort Metric.

The work-products listed in Tables 1 and 2 are taken from the Whitemarsh database project methodology. An abbreviated version of this methodology is available from the Whitemarsh website. The link is contained in the References section of this paper.

Work Unit Effort Metrics are essential when reliable and repeatable estimates are desired. Enterprise database certainly fits that class of work. The vast majority of work tasks are performed a number of times over many years to produce standard work-products. While the technology and tools may change, the fundamental work steps, effort required, and work-products do not.

This paper also sets out Hierarchical Metrics that predict the **quantity** of work units. The predictions are based on an average quantity of one work-product based on the quantity of another work-product.

This paper then defines and enumerates Work Environment Factors that directly affect the velocity of work completion. Work Environment Factors are "Murphy's Law" factors, if you will. A Work Environment Factor is a multiplier that is used to increase the original staff-hour quantity of work effort for a given work-product. Work Environment Factors are a way to deal wtih the "non-normal." The Work Environment Factors deal with team experience, client reviews, analyst/programmer environment, environment availability, and extent of end-user contact. When these factors are applied to project plans, the process of estimate development is quick.

Taken together, the Work Unit Effort Metrics, Hierarchical Metrics, and Work Environment Factors enable an estimate of the quantity of time that staff members will take to accomplish work.

This paper concludes with a description of the Project Management System and a brief scenario of how it can be used to generate project estimates in an enterprise-wide project management environment. The Project Management System is described in detail in Short Paper 8, Manufacturing Project Plans.

2. Enterprise Database Metrics

Metrics exist for each database project work-product. A review of the abbreviated Whitemarsh work breakdown structure shows that these work-products exist individually and in combinations. A individual work-product, for example, is the data element, which is used in the definition of database table columns. Work-product collections, for example, tables, include table definitions, columns, and data integrity rules.

The various metrics in this paper have been collected and refined starting in the early 1980s. The projects have all been Whitemarsh database projects. Because technology is constantly changing, work unit effort metrics are constantly updated. For example, in the early 1980s, CASE tools, code generators, and support metadata management systems such as the Whitemarsh Metabase System were virtually non-existent. Despite technology improvements the metric, Business Function Diagram Sheet **Original Development**, has not changed because the vast majority of the effort is original research and creation activities. The effort to create the actual sketch is minor in comparison.

The metric that has changed is, however, is Business Function Diagram Sheet **Final Correction and Creation**. Prior to work accelerating tools, these diagrams were all accomplished through the use of drafting tools such as "French Curves." Today, the diagram is created from the sketch and then is merely corrected, and reprinted. Since the computer is controlling all the graphics as well as the creation of all the supporting metadata and interpage connectors, the amount of time to produce the diagram is the correction time plus 5 minutes for computer generation.

3. Hierarchical Metrics

A Hierarchical Metric is a statistic about one work-product that is used to predict the quantity of another work-product. For example, on average, there are 15 columns for every table, and, there are 5 tables for every database object class. If a database contains 40 database object classes, there is likely to be about 3000 columns. While these hierarchical metrics are correct--on average--nobody's individual database is the “average.” Some are far more complex while others are simpler. Where your project fits on this normal curve is unknown.

What must be done is to examine all the accomplished database project efforts or those efforts common to the same industry grouping, and determine these hierarchical metrics. Table 1 contains a listing of the critical Hierarchical Metrics The table contains the metric’s Id, name, description, and quantity. The quantity column identifies the metric upon which the metric under discussion has its quantity based.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
36	Mission	Every enterprise has one overall mission that is subdivided into many subordinate missions. On average there is a 1 to 5 ratio between mission and subordinate missions on each level with an average number of three levels	For a medium size business unit there are about 125 subordinate mission descriptions at the lowest level
34	Database Domain	A database domain is a data intensive description of the data required to support the most detailed level of the mission description. If the noun phrases contained in the database domain description are complex then they must be hierarchically decomposed into more refined descriptions.	Every subordinate mission description at the lowest level has from one to three database domains with only two levels at the most. Thus, there would be about 125 to 250 database domain statements
19	Database Object Class	A database object class is a policy homogeneous data structure and a set of processes that maintain the business' defined policy through a well-defined set of business-based states. A typical database object class is Employee and a part of its data structure is Job Skills. Job Skills represents a class of properties that describe some policy aspect of Employee.	Every database domain infers about 10 database object classes. Across all the database domains there would potentially be about 1250 to 2500 database object classes. Since database object classes are recorded non-redundantly, about half drop out.
108	Property Class	A property class represents a set of properties that further describe some subset of the database object class's policy. A property class cannot stand alone. It requires a context, and that is the database object class. A property class is not a real surrogate for a set of database object class data elements, Rather, it is an abstract representation of later-to-be-defined database object table columns. Thus, a property class may appear on multiple database object classes and visa versa.	Every database object class has its data structure represented through the citation of about 5 to 10 different property classes. Across all database object classes about 20% of the database object classes are repeated. A repeated property class is for example, Critical Dates which would be present in Contracts, Projects, etc.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
11	Table	A table is a third normal form data structure within the business policy domain of a database object class. The fundamental data structure of a database object class is hierarchical. It becomes network when non-primary key foreign key references to valid-value tables, for example, are considered and then factored out.	Typically, the set of property classes associated with an database object class form the initial set of tables. Analysis always causes changes. Regardless, there are an average of 5 tables for each database object class.
5	Data Element	A data element is a context independent business fact semantic template. A data element has a business purpose definition and is almost always represented in one or more database object classes through the columns of database object class assigned tables. The value set of a data element may be governed by a set of rules contained in the overall Data Element Model.	Each table contains an average of 15 columns. There are about 30 column deployments for each data element.
8	Data Integrity Rule	A data integrity rule is a business statement that must test true when no updating is taking place. For example, Birth Date must be less than or equal to Death Date, or Total Department Salary must be equal to the Sum of Employee Salary. Data integrity rules are best implemented as fundamental parts of DBMS schemas as column constrain clauses, table constraints, assertions and triggers. If these are not possible, Data integrity rules are implemented as software modules included with every appropriate Add, Delete, or Modify routine.	There are an average of 5 data integrity rules for each table.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
22	Database Object Table Process	A database object table process is a primitive data transformation that causes changes to the database. There are three types: add, delete, and modify. A database object table process either completes or fails. A database object table process adds, changes, or deletes only one row. If it fails then the change to the database reverts to its original state.	There are three primitive data transformations for each database table.
102	Database Object Information System	A Database Object Information System is a database process that changes values in the database. Each database process is either primitive or composite. A primitive database activity, called a primitive data transformation. A composite database process is one that decomposes into a set of primitive data transformations.	Not counting the primitive data transformations, there is at least one database process for each data integrity rule that cannot be encoded directly into the schema as a referential integrity rule & action.
101	Business Event	A named and described instance of a use of a business information system. As business functions are accomplished and require services from a business information system, the business event occurs and causes the execution of a business information system.	There are as many business events as there are uses of business information systems.
52	Business Function	A business function is a named set of human activities that may involve other business functions and people. If data is transformed, that is accomplished by use of a business event that signals the execution of a business information system.	Business functions are hierarchical. There are probably the same number of business function tops as the maximum number of subordinate missions. There after, the business function spreads out three to five to a level for about four levels. If there are 125 subordinate missions the estimated business functions would be about 3,400.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
72	Business term	A business term is a name that has a special meaning to the database system. The term may be subject matter or computer related. A subject matter related term might be, for example, remaining balance. A computer term would be DBMS, the abbreviation name for the database management system.	An average business MIS contains in the neighborhood of 250 special business terms.
53	Computer Module	A computer module is an occurrence of computer code contained that accomplishes a specific objective. If the computer module is also a stand-alone computer program then the program's language may be natural, O/S command language, or compiler-based language such as COBOL, C, C++, Visual Basic or Clarion. A computer program typically contains one or more modules which interact with data from one or more views. While collections of modules comprise computer programs, the actual quantity of computer programs can be quite arbitrary. For example, a PC based system may consist of one computer program that in turn contains up to a hundred discrete modules.	There is at least one computer module for each database process. In addition there would be at least one module for each screen, each report, each file operation and for each programmed data integrity rule.
103	DBMS Schema	The DBMS schema is a linguistic expression of the syntax and semantics of the interface between the database designer's concept of the database and the DBMS. The rules specified in the logical schema apply to all users of the database.	There is at least one DBMS schema for each instance of data under control of an executing DBMS. Some DBMSs can control multiple DBMS schemas.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
15	DBMS Table	A named collection of columns of data that is under the control of a specific DBMS and that resides within a particular schema.	A DBMS schema typically contains from 40 to 4000 tables depending upon whether the DBMS can control one or more than one DBMS schema. The fewer the number of controlled schemas (1?) the more the number of contained tables. The actual quantity of DBMS tables depends on the number of data tables that are retained in third normal form or that have been transformed from third normal form. A good rule of thumb is that the quantity is the same as data tables.
97	View	A view is a business information system schema object interface mechanism of the DBMS Views consist of view columns. A view column is mapped to one or more compound data elements, or one or more derived data elements, and one or more DBMS column. A view column also forms the basis for the interrelationship among views.	The minimum quantity of tables would be the quantity of computer programs. There could be as many views as there are DBMS tables.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
106	Entity Relationship Diagram (ERD)	A diagram that illustrates entities, and the relationships between entities. In the Whitemarsh methodology, there are three classes of entities: Data elements, property classes, and database object classes. Entities within the ERDs often result tables (a proper meta-object within the Metabase System's Implemented Data Model. Other entities result in data elements, a proper meta-object within the Data Element Model. The third class of entity results in Database Object Classes, a proper meta-object within the Database Object Class model. Within the Whitemarsh methodology, entities are therefore an intermediate analysis work-product that upon further analysis become data elements, tables, or database object classes.	There are as many entity relationship diagrams as database domains. Once all are drawn, then there could be several levels of "higher" diagrams.
105	Entity	An entity is a unit of data from within the ERD that is considered important by an analyst. In the Whitemarsh database project methodology, an entity is not intended to be well defined or precise. Rather, an entity instance (Company, Salary, Contract, Employee Education) merely becomes a data requirement that must be handled by the database application. An entity may transform itself into an database object class if and only if it "passes" certain tests. Entities that are not database object classes may become property classes (represented as Implemented Data Model tables of policy homogeneous columns) or a data elements.	Each entity relationship diagram commonly contains about 10-15 different entities. Many of these entities exist in common across all the entity-relationship diagrams. Thus, while they are named redundantly, they are defined non redundantly.

Hierarchical Metrics			
Metric Id	**Name**	**Description**	**Quantity**
107	Business Information System	A named collection of data processing activities that may involve the use of internal processes, database processes, reports, files, and screens. The information system may have subordinate information systems. The contained logic is cast into information system control logic blocks and subordinate information system control logic blocks.	The quantity of information systems is initially determined by counting the nodes from the sets of Resource Life Cycles. Since there are typically 15 Resource Life Cycles s and about 10 nodes on each Resource Life Cycles then there would at a minimum be 150 information systems. That number is likely to grow through a decomposition of Resource Life Cycles nodes. The maximum number is probably about another 33%, or 200.
25	Screen	A display of ANSI strings for labels and graphical divisions, messages, and data entry positions.	The quantity of screens is generally equal to the number of primitive data transformations.

Table 1. Hierarchical Metrics.

4. Work Unit Effort Metrics

A Work Unit Effort Metric is an estimate of the quantity of staff hours it takes to accomplish a single unit of the identified work. The Work Unit Effort Metrics are listed in Table 2. Each work effort metric consists of its identifier, a short descriptive name of the metric, a description of the metric's purpose and target, and then the quantity of staff hours required to accomplish the work. The work-products from which these Work Unit Effort Metrics come from the Whitemarsh methodology.

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
50	Access strategy creation, review, & revision per table	Time necessary to construct an access strategy as a prelude to encoding the actual implemented data model statements for the DBMS.	1.10
63	Business function--intermediate specification	Time necessary to take the high level business function and develop the next level or two of detail, stopping before any detailed partitioning and pseudo code development.	4.40
62	Business function--high level specification	Time necessary to create high level business function text and diagram per event, once all the mission descriptions are fully known, reviewed, and approved.	8.80
101	Business event identification	Time necessary to identify, name and describe an instance of a use of an information system.	.5
1	Business function diagram sheet development	Time necessary to create a graphic that depicts from three to seven different business functions, associated views, and all appropriate labeling.	4.40
46	Business function review and revision	Time necessary to review a particular business function, including the business function, specified views, database processes, and one level of subordinate business functions. Changes would be minor rather than major.	4.40
100	Business function diagram sheet final correction and creation	Time necessary to create a final graphic that reflects the final, approved business function. All corrections are very minor	.25
52	Business function--detailed specification	Time necessary to create programmer level pseudo code per module.	17.60
72	Business term definition per term	Time necessary to generate a clear definition	.17

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
74	Business term--review and revision per business term	Time necessary to present the business term to a group of knowledgeable users and to make minor changes.	.11
90	Client review time of a deliverable	Amount of time that the client desires to review a deliverable. This is clock time. Thus, if there is a 80 review and if 2 staff are waiting until the review is complete, then the quantity for this metric would be 160.	1.00
89	Computer unavailability time effect	Time to be allocated into a computer intensive effort when ever the computer is not available. If a metric takes 8 hours, and the computer is available 50% then quantity is 4 hours times the quantity of the 8 hour metric.	1.00
86	Data elements--external view analysis and definition	Time required to identify all report, screen and internal external view elements. Store, sort, & define the unique set with a one line description, data type, class, and picture. Missing info is to be provided by the client.	.11
4	Data conversion elapsed time for each table	Elapsed time to accomplish data conversion. That is, execute, check out the database, and develop a report indicating the accomplishment.	35.00
5	Data element definition	The total time for all research, forms analysis, etc.	2.20
6	Data element quality review	Time necessary to present the definition of a data element to a group of knowledgeable persons and to receive feedback for specific changes that might be necessary.	.90
41	Data element review and revision	Time necessary to present a data element definition to a knowledgeable committee and to receive changes requests. Time also for accomplishing the change in preparation for a data element report or for the next review.	.11

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
8	Data integrity rule specification	Time necessary for the actual research and construction of an average data integrity rule, not including the specification of keys, or edit tables, as they are covered by other metrics	1.10
7	Data integrity rule review	Time necessary for the presentation of a specific data integrity rule to a knowledgeable committee and to receive requests for changes.	.18
58	Data integrity rule review and revision	Time to review the definition and correctness of a data integrity rule.	.11
9	Data loading subsystem development time per table	This time includes time to accomplish the detailed design, coding, and unit testing from a set of well defined specifications.	44.00
42	Data model--specified review & revision per table	Includes review and revision from mission descriptions, database domains, database object classes, property classes, tables, data elements, data record elements, keys, edit table and data integrity	2.20
73	Data model--specified to implemented mapping per table	Time necessary to take a specified data model table and create the binding strategy for mapping to an implemented data model table.	1.10
2	Data storage definition language creation for each table	Time necessary to construct the actual syntax for the access strategy clauses once the access strategy is completely defined.	.28
34	Database domain development	The amount of time to create these work-products for each subordinate mission description. This includes the development time, presentation to the user, and final work-product preparation.	26.40

Work Unit Metrics			
Metric Id	Metric Name	Metric Description	Unit Effort
102	Database process creation	Time necessary to fully describe in pseudo code a an activity that changes values in the database. Each database process is either primitive or composite. A primitive database activity, called a primitive data transformation. A composite database process is one that decomposes into a set of primitive data transformations.	1.00
103	DBMS schema development	The time necessary to prepare and submit the DBMS schema to a DBMS.	1.00
104	DBMS table definition	The time necessary to prepare and submit a table to the DBMS. Included are definitions of all named collection of columns of data that is under the control of a specific DBMS and that resides within a particular schema.	1.00
3	DDL creation time for each table	Time necessary for the creation of the implemented data table clauses for each specified data model table. Time includes schema, data record, data record element, relationship, and necessary integrity clauses.	2.20
13	Documentation (database) development	This is the time required to accomplish the table section of the logical database report.	17.60
95	Documentation (operations) development	Time necessary to develop suitable documentation for a computer center operations staff. Time is for the development of each self contained operation that the data center must perform. Time includes walk through with center.	2.20
12	Documentation (system) development time	This time is for the complete documentation of each module. A program may be a module or consist of more than one module	8.80
57	Documentation (system) review and revision	Once submitted and reviewed, the time necessary to complete changes to documentation. The requested changes are only light editing, spelling, grammar, etc. No major rewrites.	13.20

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
94	Documentation (user manual)--review and revision	Time necessary to review and revise an existing and complete user manual. Time is for each screen and/or report.	2.20
14	Documentation (user manual) development	This is the time required to accomplish the development of the documentation of a user manual for each screen or report defined in the manual.	17.60
60	Documentation files setup for task	The time necessary to make labels, get file folders and to arrange subtask folders for all deliverables within a task.	2.20
30	Edit table definition time	The time necessary to accomplish the definition of an edit table's set of values. The loading, maintaining, etc of that table would be accomplished through a program with only a few modules.	4.40
65	Edit table review and revision	Time necessary to review and create revisions to a specific edit table, including its name, definition, and various values.	1.10
105	Entity identification and definition	The time necessary to research documentation supporting the defined database domain and to identify the noun phrases as entities. Included also is the amount of time to define the entity to the metabase and to create two different relationships between the entity and surrounding entities.	.50
106	Entity relationship diagram development	Time necessary to take a defined set of entities from within a database domain and to create an entity-relationship diagram. Included also are all the named relationships.	2.00
82	File (computer) analysis of file elements--per file element	Time necessary to review a specific file element, its name definition, picture clause, and to related it back to a data record element. This metric assumes the existence of sufficient documentation to accomplish the mapping.	1.10

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
81	File (computer) analysis time	Time necessary to review the documentation surrounding a computer file and to identify the specified data model tables that relate to the file's contents	8.80
107	Information system identification and definition	Time necessary to identify name and describe each discrete collection of data processing activities that comprise a specific information system. An information system may involve internal processes, database processes, reports, files, and screens. An information system may have subordinate information systems. The contained logic is cast into information system control logic blocks and subordinate information system control logic blocks.	4.0
43	Logical database specification report per table	Time necessary to bring together all the material to create a complete report regarding the specified data model for a database. Report includes materials from mission description through to data record, data record element, keys, etc.	.28
47	Logical database diagram creation time per table	Time includes the original sketch, and then drafting the first work-product using a PC based graphics package. Notation is limited to one-to-one and one-to-many, with all many-to-many reduced to two one-to-manys.	.22
40	Metabase component instance review and revision	Data element, database object class, data record element, database process, data integrity rule, business function, mission description, etc.	.55
51	Metabase component instance final review	After all definition, installation, and walk throughs of metabase component instances, a final review is performed before a work-product is turned into the client.	.11
39	Metabase component instance review time	Mission description, database domain, business function, view, database process	.28

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
16	Metabase entity attribute value entry	This is the time for entering the values for all the attributes values for each entity instance. This presumes that the attribute values are already known.	.04
15	Metabase entity instance data entry	This is the time necessary to accomplish the data entry transaction for each entity type's instance. This presumes that all the values are known for the entity instance.	.04
17	Metabase relationship instance entry	This is the amount of time necessary to accomplish the entry of the values for each relationship instance	.04
66	Mission description diagram review and revision	Time necessary to review the mission description diagram and to create specific changes that are readily available from those conducting the review.	2.20
36	Mission description diagram development	The time necessary to accomplish the development of the mission description diagram once all the subordinate mission description diagrams are developed.	26.40
18	Module development	This is the time to accomplish the development of a module, that is, its detailed design, coding, and unit testing. A module is a process unit that is in third normal form. A program may be a module or it may contain many modules.	44.00
19	Database object class definition	The amount of time necessary to review the entity relationship diagram, to identify an database object class, and then to define it.	1.10
20	Database object class review	The amount of time to accomplish a review of the quality of an database object class.	.11
67	Database object class review and revision	Time necessary to review and create revisions to the specification of an database object class.	.22

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
44	Phase review per subordinate Mission description. Develop and conduct	Time necessary to create the materials necessary for the full review of a subordinate mission description. The review would start at the subordinate mission description and continue through the appropriate data and process	159.00
70	Physical database specification report per business function	Time necessary to bring together all the materials to create a report regarding the specified process model, physical database characteristics, etc.	.28
59	Presentation per subordinate mission description--creation	Time necessary to put together, dry run, and finalize a presentation for a subordinate mission description, given that all the materials already exist in the metabase, and can be brought together quickly.	26.40
22	Primitive data transformation specification	Time necessary to create the pseudo code for each database process. A database process is single purpose, that is, in third normal form.	4.40
68	Primitive data transformation review and revision	The time necessary to review the pseudo code that represents the primitive data transformation.	.28
69	Process model--specified, review & revision per business function	Includes mission descriptions, business functions, views, primitive data transformations, edit tables, screens, files, reports (computer), data integrity rules, and mapping to the specified data model through data record elements.	4.40
21	Project phase planning hours	The time necessary to identify the Whitemarsh tasks, develop an estimate for a PERT, Gantt, and CPM within a PC base project management package.	44.00
84	Project phase planning hours--review and revision	The amount of time necessary to perform minor revisions to a project plan that was submitted as part of a proposal.	17.60

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
108	Property class identification	The amount of time necessary to identify that an entity is in fact a property class. Included is the property class's name and brief description.	.5
91	Prototype presentation session time	Time to conduct a prototype session of a number of screens and reports. The session is half, and the preparation time is the other half. If the session is multiple days, then multiples of the metric	8.80
71	Report (computer)--detailed external specification & design	Derived report elements, and all formulas. Includes mapping back to database data record elements.	17.60
85	Report (computer)--summary specification and design	Time required to perform a surface analysis of a report only after the layout has already been done. Determine external view elements, selection & sort criteria from rpt's view & detail the formulas. The output is not a programable spec.	6.60
56	Report (computer) revision if developed in natural language	Time necessary to revise a simple report through a natural language. Time includes revision of its design, coding, several sets of prototype runs, and demonstration to the customer for review	2.20
55	Report (computer) simple development thru a natural language	Time necessary to generate a simple report through a natural language. Time includes design, coding, several sets of prototype runs, and demonstration to the customer for review	4.40
77	Report (computer) element analysis and definition time	Time necessary to identify, analyze, define, and determine the appropriate formulas for generating a particular cell on a report. Includes also the identification of any edit tables necessary for data record element. Value transformation	1.10

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
24	Report (computer) development	The amount of time necessary to accomplish the development of a single, simple report. A simple report is one that retrieves data from a few tables and performs simple calculations. Work: detailed design through unit test	26.40
23	Report (computer) analysis time	Client provided report format and develop an understanding And also a brief, several sentence description.	2.20
29	Report (computer) testing	The amount of time to test a report that has been developed.	8.80
54	Report page, development, review and final for deliverable	Time necessary to create a report that is to support a work-product for delivery to the client.	2.20
88	Report page: Assemble already drafted and reviewed material	Time necessary to take already assembled, stored on micro material and bring together a document. Includes all headers footers, draft printing, editing and then transmittal to customer.	.22
61	Review committee setup for a task	Identify the names, phone numbers, and addresses of those established to review Whitemarsh interim work-products and deliverables.	8.80
83	Review--formal clock time for conducting a review--by reviewer	Staff hours for holding a review in front of the client or an internal review. For a three day review there would be 6, and if 3 staff were involved the number would then be 18.	4.40
25	Screen (computer input) analysis	The amount of time necessary to take a client provided screen and to identify the appropriate set of external view elements. Also time necessary to determine appropriate editing and validation (computed	4.40

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
93	Screen (computer) review and revision	Time necessary to perform revisions to a screen that is already been designed, programmed, and tested. Includes time for documentation revision, and testing of changed screen.	4.40
92	Screen (computer) design and development time	Time necessary to use a screen generator to develop a screen from an already prepared screen specification. Time includes design, PF keys, messages, and navigation to lower screens as appropriate.	8.80
78	Screen element analysis and definition	Time necessary to identify, analyze, and define the screen element, including its characteristics (length, size, picture, etc.) And all edit and validation tables necessary to support it.	1.10
37	Security definition time per view	The amount of time to identify the security requirements per view and to record that into the security requirements entities in the metabase.	4.40
87	Security time per report and/or screen data element	Time necessary to state the security restrictions imposed on a screen element or a report cell. Time also to store security, sort, and report by screen, report, and screen/ report data element. Missing info is provided by client.	.06
76	Security--user id. & Acquisition of password, per password	Time necessary to identify a user and to fill out the paper work to submit a request for a password.	2.20
53	Software unit delivery from development/test to production	Time necessary to move a specific software unit (program) from one environment to another.	.55
35	Subordinate mission description development	This includes the amount of time necessary to review materials that are adequate, sit in a user presentation of the subordinate mission requirements, develop the subordinate mission description, present it to the user, and finalize	26.40

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
80	Subordinate mission description user guide analysis-partial	Time necessary to analyze a major section of a subordinate mission description user guide and report on its completeness the report includes identification of the missing or incomplete sections.	8.80
79	Subordinate mission description user guide analysis-complete	Evaluate the completeness of a subordinate mission description user guide and provide a written report on its acceptability, including the specific areas or components that are missing.	26.40
45	Subphase review per subordinate mission description, development only	Time necessary to develop all materials necessary for a review of a specific subordinate mission description, e.g., budgeting. Review starts with subordinate mission description and continues through the data and process models.	88.00
75	Subsystem design--update, loading, or reporting per business function	The amount of time necessary to collect the already defined business functions, views, database processes, etc. into appropriate subsystem designs.	2.20
11	Table development	Includes definition, data record element allocation, keys, edit tables, presentation, and first review.	8.80
10	Table review	Time necessary to present the name, definition, data record elements, and keys to a knowledgeable committee of reviewers, and to accept suggestions for changes.	.17
64	Table review and revision	Time necessary to review and make revision to the spec. Of a table, including its data record elements, primary and foreign keys, and edit tables.	.55
26	Test data creation per table	The amount of time necessary to create realistic test data for each table. It must test all boundary conditions and all data integrity rules.	17.60

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
48	Test data creation for module testing	The time necessary to review the logic of a module and develop sufficient test data to test every branch and/or if statement, and every program based edit and validation. Data is on paper and/or in computer files for program use.	22.00
27	Testing per data loading system	Time necessary to accomplish a test of a data conversion program.	22.00
28	Testing per module	The amount of time allocated to accomplish testing for each module.	22.00
96	Testing--system quality test, planning and execution	Time necessary to plan and conduct a single instance of a complete system quality test. Time estimate is for each screen, batch update program, or report that is to be produced. Half of the time is for planning and half for conducting system quality test.	2.20
98	Testing--system quality testing for update subsystem	Actual time required to perform testing for an update subsystem after system development testing is completed. Computed on the basis of the number of programs involved in total testing effort. If subsystem has 30 programs, then 30 is quantity.	26.40
99	Testing--system quality test for reports subsystem	Time for testing of a complete reports subsystem after the system development testing is completed. Time is for testing each program involved in subsystem. Thus, quantity is the total number of report programs involved in subsystem.	16.50
32	Training development time per class hour of delivery	The amount of time necessary to develop user manual based training. The time includes outlines, instructor notes, and appropriate computer demonstrations to allow students to understand the system's operation via the user guide.	8.80
30	Training (maintenance) class delivery time per table	This time is for all activities related to the table, that is, the logical design, data conversion, update, reporting, and system control.	35.20

Work Unit Metrics			
Metric Id	**Metric Name**	**Metric Description**	**Unit Effort**
31	Training (user) class per instance per subordinate mission description	The amount of time necessary to deliver the training for each subordinate mission description to a set of users.	35.20
33	View development time (specified)	The amount of time necessary to create the view that is required in the specified process model.	2.20
97	View development time (implemented through a generator)	The time necessary to accomplish the generation of an implemented view through a view generation program. The specified view must have already been accomplished and stored in the metabase for the generator to access.	1.00
49	Walk-through for module by development team reviewers	Clock time necessary for a walk through of a module. Walk-through starts with purpose, design, graphics, pseudo-code, program listings, test data, and testing plan.	4.40

Table 2. Work Unit Metrics.

5. Work Environment Factors

Work Environment Factors are multipliers. They are related to the performance characteristics of workers in the work environment. Similar to Work Unit Effort Metrics, the values associated with the Work Environment Factors is an estimate based on experience. The experience of your team in your environment with your tool set is likely to be different. As above, an analysis should be made of your projects et al and the values in Table 3 should be adjusted.

The Work Environment Factors are cumulative. That is, if there is a multiplier of 1.2 combined with one of 1.4 then the result is 1.2 times 1.4, or 1.68. The factors were identified through observations of the skill levels of various workers using different classes of tools. It is critical that when factors are employed that very good statistics are kept so that these factors can be updated through the actual experiences of your organization. The types of factors are:

- Experience of team
- Reviews conducted by client

- Programmer/analyst tools environment
- Environment Availability
- Extent of user contact

Work Environment Factors.

Factor Name	Factor Value	Factor Description
Experience of Team	1.00	No effect
	1.00	Competent team that has performed this type of work before
	1.10	Experienced DP staff but novices at this type of work
	1.20	Novice DP staff
Reviews Conducted by the Client	1.00	No effect
	1.00	Reviews, that is, walk throughs are conducted by the client as scheduled
	1.10	Reviews are conducted but by inexperienced reviewers
	1.20	Reviews are generally not conducted
Analyst/programmer Tools Environment	1.00	No effect
	0.75	Work station with case and metadata management system.
	0.60	Work station with case and metadata management system, and shared metadata system.
	0.40	Work station with case and metadata management system, and shared metadata system and code generator.
	1.05	Work station with only word processing and stand-alone case tool environment.

Work Environment Factors.		
Factor Name	**Factor Value**	**Factor Description**
	1.10	Work station with only word processing and stand-alone “visio” tool environment.
	1.25	Work station with only word processing and no case/repository environment
	1.30	For server only terminals that only provide access to a mainframe word processor
	1.40	For no equipment available to the staff, except through an administrative person
Environment Availability	1.00	No effect
	1.00	If the equipment and all required software is available for at least 6 business hours each day
	1.16	For each hour below the average of six that the equipment is unavailable
Extent of User Contact	1.00	No effect
	1.00	If the users are available within half day request to review interim work-products (work in progress, but not a deliverable).
	1.10	If users are available but only within 2 business days of request
	1.20	If users are available but only within 4 business days of request
	1.30	If users are available but only within 6 business days of request
	1.40	If users are generally not available

Table 3. Work Environment Factors.

6. Project Management Database

Correct project estimating is critical. In any estimate there are three components:

- What is done.
- The quantity of what is done.
- The work environment under which the work is done.

The first component relates to the work breakdown structure (WBS). The critical question is whether the WBS is sufficient, and correct. If the WBS does not call for the correct steps or calls for an insufficient number of steps, an accurate estimate can only be an accident.

The second component of an estimate relates to knowing the quantity of work that has to be accomplished. If the work is well partitioned, then, over time, critical Hierarchical Metrics can be developed. For example, for each subject area database there are—on average—200 tables. Each table contains on average 15 columns. And, on average, there is about 0.03 data elements per column. Thus, given the Hierarchical Metric of how many subject area databases, there are about 100 data elements. From this example, whole estimates can be developed from knowing only key component volume counts and the various *on-average* Hierarchical Metrics. The amount of time is determined by multiplying the unit effort by the quantity of units. If it takes 4 hours for a data element's definition (including all research and valuation of upper meta-object levels in the Metabase’s Data Element Model), then the time is about 400 hours.

The ratio of data elements to columns, 1:30, is very dependent on the quantity of database projects. For the very first project, the ratio is likely 1:5. That produces about 600 data elements. As more and more projects are accomplished, the greater will be the reuse of data elements. The ratio will then become smaller and smaller from 1:5 to about 1:30.

The third component relates to the set of Work Environment Factors under which work is being accomplished, for example, the availability of productive CASE tools, Workstations, LANs, software generators, and the like. If the work environment is less than ideal, the unit effort must be multiplied by a Work Environment Factors greater than 1.0. If the Work Environment Factor is better than “normal,” the Work Unit Effort Metric is multiplied by a fraction.

If the final factor is 1.5, then the estimate of 1500 hours for data element definition must be raised to 2250 hours.

The objective of any estimating process is to create an estimate that is reliable, repeatable. If it exhibits these characteristics and is stored in a Project Management System. It’s database engineered. Thus, it can hold project plans across the enterprise, can support where-used, built-by, and many other kinds of reports, and its base-line metrics, whether Hierarchical Metrics, Work Unit Effort Metrics, and Work Environment Multipliers can all be updated. More importantly, Project Management System estimates, overtime, become part of the project management lessons learned that guide development.

Estimating Errors come from two sources: scope and familiarity. As more enterprise database projects are accomplished, the kind of work becomes better known. During this initial time period, it is common for an organization to underestimate their own business's complexity. Unknown or undiscovered policies and procedures slither out from under every rock and cause surprise requirements. But as time goes on, the estimating-staff better understands the nature of the company's business. Better scope of work estimates can be done. As time goes on, the Work Unit Effort Metrics, Hierarchical Metrics, and Work Environment Factors metrics behind the estimates become more accurate because users of the WBSs understand the process better and the work-products become more familiar.

The Project Management System is explained in the #12 Whitemarsh Short Paper, Manufacturing Project Plans. The data model of its database is presented in Figure 1. The entities from Figure 1 are also divided into six distinct clusters, which are:

- Contracts, organizations and contract [staff] resources
- Resource and Resource Life Cycle Node
- Project, Deliverable, and Task Templates
- Project Staff
- Project Building and Estimation
- Project Work

In general, the *Contracts, Organizations, and Contract [staff] Resource* cluster of entities represent the environment within which projects take place.

The *Resource and Resource Life Cycle Node*[1] entity represents the target of the project, that is, the area of the business benefitted by the project. For example, for manufacturing, finance, human resources, or land use planning.

The *Projects, Deliverable, and Task Templates* entity cluster enables the definition of the templates employed in the actual building of projects. Defined across the enterprise, these templates enable standard project execution and accomplishment measurement.

The *Project Staff* entity cluster enables the inclusion of the staff as resources for a contract, and also permit the specification of the specific types and performance ratings of skills that a person may bring to a specific project.

The *Project Building and Estimation* entity cluster represents the entities that support building projects. Projects and associated tasks are initially created through the use of the Project Deliverables, and Tasks Templates. Once projects and associated tasks are created, they are modified by attaching work environment factors and specific skill-level based staff assignments. Only then can task and project resources be computed.

Finally, as task work is accomplished, the *Project Work* entity is valued. As actual work is accomplished, it can be reported through any of its related entities.

[1] Resource and resource life cycle node has the exact same definition as it does within the Whitemarsh metabase and also the process of Resource Life Cycle Analysis of Ron Ross.

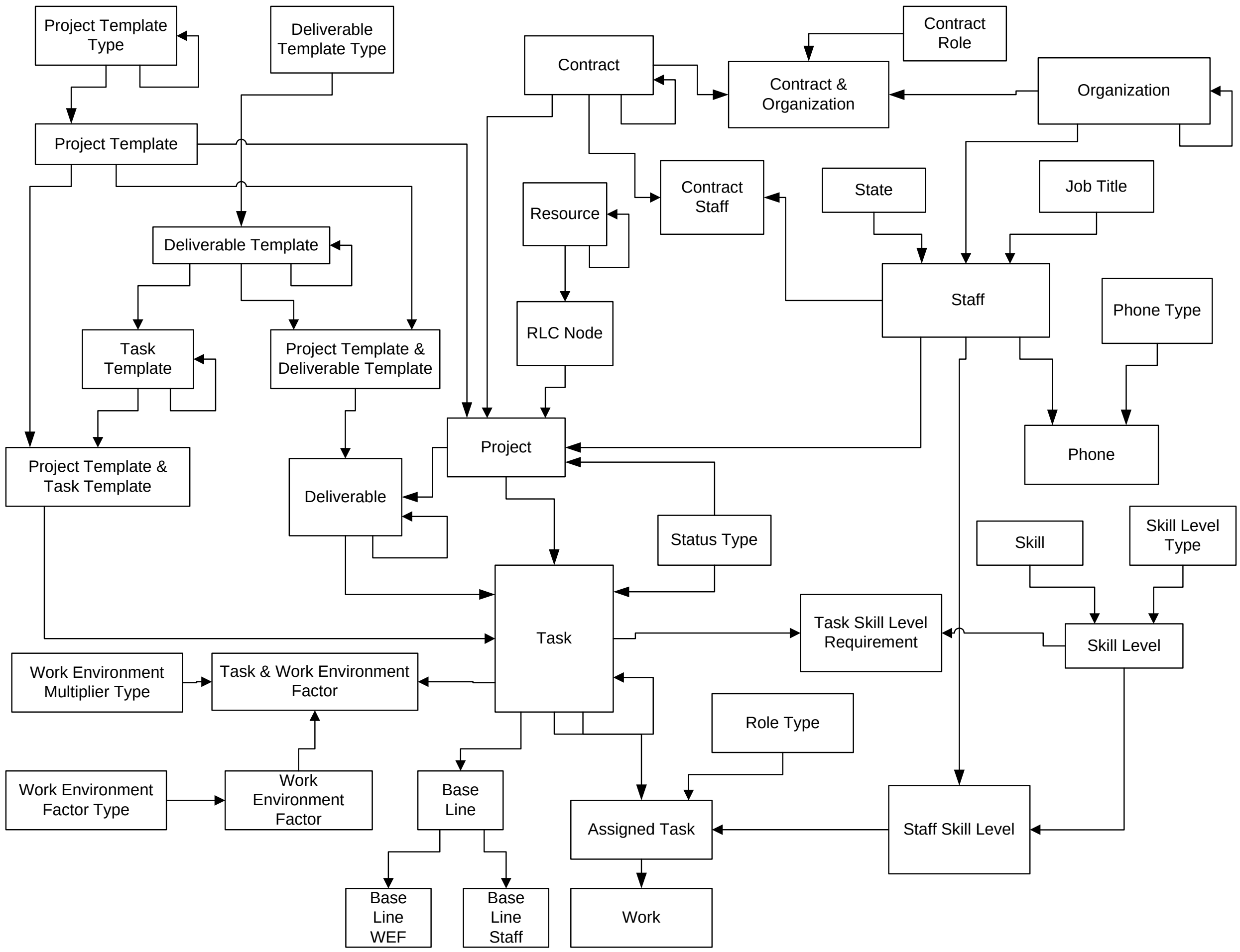

Figure 1. Project Management System database model.

To create a project, the following steps must be accomplished:

1. Create the project record
2. Choose the project template
3. Choose the project's status
4. Choose the project's resource
5. Select the project's contract
6. Enter the project's description
7. Choose the project's leader
8. Create basic information for each task
9. Choose the appropriate work environment factor for each task
10. Assign the tasks to project staff members

11. Choose the role that each project staff member performs on a task
12. Press the Compute Project Resources button on the project record
13. Record the project's base line

In Step 1, the project record is created on the screen that contains the list of existing projects. Upon insert, a new [but null] project is created and is presented for further update.

Step 2 requires the selection of the project's project template. A set is presented and one is chosen.

Step 3 includes supplying the project's title, and a status that is chosen from the status type entity list.

Step 4 sets the project within the context of a particular Resource Life Cycle Node. Thus, it is related to a specific collection of databases and business information systems, and exists within the overall life cycle of the Resource.

In Step 5, the appropriate contract under which the task is performed is selected.

In Step 6, a 250 character description of the project can be included. While this seems short, it must be remembered that there are many other name and description attributes that in combination serve to fully describe both the project and its context. Included for example are all the deliverables and their templates, tasks and their templates, contracts, and resources.

In Step 7, the project's leader is chosen from a staff list. At this point, the only additional piece of information that can be added at the project level is the project's start date. Once this is entered, all remaining work must be done at the individual task level. This is done in steps 8, 9 and 10.

Once the project's basic information is captured, the project's basic information is essentially complete, and the auto-build of the project can start. Auto-build is the manufacturing step. Project classes are identified, defined, and include deliverables, and unit effort. Once refined, these projects are made into templates. Thereafter, these Project Template are available to manufacture projects.

Auto-build is accomplished by highlighting the newly created project and instigating the project build process. Immediately, all the deliverable template records and task template records associated with the identified project template record are copied into the project's deliverable and task entities. At that point, the lists of deliverables and tasks can be displayed

Once the project's auto-build step is complete, the project's tasks can be updated. This process, that is, Step 8, is started by highlighting the project from within the project list and instigating a change. The project's update window is then presented. Each project task can be changed by first highlighting a specific task in the project's task list window and invoking the change function.

In Step 9, the appropriate work environment factors that govern the tasks are picked. Once picked, the task screen shows chosen work environment factors. When a new work environment factor is needed, an exiting set is presented. Choices should be made that relate to the work environment for that specific task. Each chosen work environment factor has an its effect on the unit effort hours computed for the task.

The staff hours in the task templates assumes normal work environments for each task. While the definition of the work environment factor types and the specific factors is clearly under the control of the project management administrator, Table 3 provides a good initial set.

In Step 10, staff members are assigned to the task. The list of currently assigned task members appears on the task screen. Changes in these assignments can be accomplished through inserts, changes or deletes.

On an insert, a list of staff members and their associated skills is presented. When a specific staff member is chosen, so also chosen is the staff's skill and skill level multiplier.

Once a staff member has been assigned to a task, Step 11 consists of picking their appropriate role. Suggested roles might be:

- Developer or contributor
- Manager or leader
- Reviewer

Once all the work environment factors and staff have been assigned to a task, the resources associated with that task can be computed. This is done through three processes:

- Compute Factored Hours
- Compute Elapsed Hours
- Compute Completion Date

When the Compute Factored Hours process is executed, the standard hours from the task's task template is multiplied by the multiplicative summation of all the work environment factors producing Factored Hours.

When the Compute Elapsed Hours process is executed, the staff work hours for each assigned staff member is computed as the Factored Hours multiplied by the staff's Skill Level Multiplier. The elapsed hours for a task is determined as the longest effort for any one participant in a task. The field, staff assigned hours, is the commutative quantity of all staff work hours.

When the Compute Completion Date process is executed, the start date of the task is employed to then compute the completion date according to these rules:

- Duration Days are computed by dividing elapsed hours by hours-per-workday. The Whitemarsh project management software's basic reference data can be changed. It includes, for example, hours-per-workday. If a task has a duration of 24 staff hours and if the hours-per-workday are 6, then the staff-days is equal to 4.

- The total staff-hours for a project is the sum of all the staff hours.

- The completion date depends on whether a task which contains other tasks is marked as serial or parallel.

  If the containing task is marked as serial, the completion date for its contained tasks is computed to be start date for the first contained task plus the duration days for every other contained task. Weekends and holidays are excluded.

  If the containing task is marked as parallel, the completion date for its contained tasks is computed to be the start date for the first contained task plus the duration days for the longest contained task. Weekends and holidays.

In step 12, the overall project resources and end-dates are computed. Once all the changes to each task has been completed, the Compute Project Resources process is executed. At that point, the project's factored hours, staff hours, and elapsed hours are all computed. The start-date and end-dates for each task are recomputed beginning with the project's start date.

Unless otherwise indicated, tasks within a project are processed in a hierarchically serial manner. If a project contains multiple tasks, or if a task contains other tasks, and the schedule-parallel indicator is parallel, the contained tasks within the project or within specific tasks are scheduled in parallel. That is, the elapsed time for a containing project or task is the elapsed time of the longest contained tasks. That set of tasks becomes the critical path. Once all the scheduling is accomplished, the project's end date is computed.

Once a project is scheduled, the Check Overload process can be executed. This process triggers an analysis that determines if any staff member is overloaded for a task. It determines this by summing the staff members assigned hours across task schedules. If the sum is greater than the allowed work hours for the period, then the overload indicator is set. A staffing allocation report is generated and it shows the project and task level of staff loading by staff person across time. The granularity of time used to assess overloading is one week.

Step 13 enables the project manager to record a baseline for the project. A baseline is the dated permanent recording of key project, task, resource and schedule information. When the baseline is recorded, a new set of the information is created for every task in the project. Collected as well is the key information from the work environment factors and the staff assignments on the date of the baseline.

Once project planning is finished, the project's baseline is created by invoking the Baseline process on the project update window. The data values for all of the above attributes are collected on a task by task basis and stored into the baseline entity. Baselines support progress reporting. Since a key attribute of the baseline entity is the baseline date, multiple baselines can be created and the baseline information can be described and tracked during the life of the project.

7 Statistics Capture

The real value of Project Management System is the capture of ongoing statistics, and the reporting of earned value. As projects are accomplished, time cards need to be created that record both the time spent on a WBS task and the quantity of worked units. For example, if the Project Management System states that the amount of hours for a data element's definition is 2 hours and the environment factor is 1.75, and if through time card tracking the amount of time—on average—per data element is greater than 3.5 hours, then either the metrics or the environment factors need to be adjusted. Over time each organization develops its own valid set of metrics and environment factors.

When the promises of proposals meet the work accomplishment of time cards, a statistic called earned value can be produced. Earned value simply is a measure of the amount of *funds* that have been earned based on work performed. It is common for projects to not really know what their earned values are because work performance is seldom tied to time expended.

8 Conclusions

There are three key points about enterprise database estimating. First, virtually all initial projects are estimated wrong. If the Project Management System captures performance-based statistics there can be continuous adjustments to the Work Unit Effort Metrics and the Work Environment Factors. Second, all early-on enterprise database projects should last less than one year and involve no more than about ten staff years. This enables quick results for relatively small amounts of money. Third and finally, the repository is absolutely essential for enterprise database. It enables all project personnel to be aware of everybody else's progress and evolving corporate semantics.

9. 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:

Paper	URL
Paper 11: February 2008 - Engineering and Managing Information Systems Plans Paper 12: April 2008 - Manufacturing Project Plans Paper 13: June 2008 - Function Points Strategy for Business Information System Estimating	http://www.wiscorp.com/short_paper_series.html
Information Systems Planning: Book, Course, and Presentation (short and long) – samples Knowledge Worker Framework: Book, Course, and Presentation (short and long) – samples	http://www.wiscorp.com/EnterpriseDatabase.htm
Database Architecture Classes: sample	http://www.wiscorp.com/DatabaseDesign.htm
Resource Life Cycle Analysis: Paper	http://www.wiscorp.com/MetabaseProducts.htm
Database Project Work Breakdown Structure – sample	http://www.wiscorp.com/DatabaseProjects.htm
Information Systems Planning: Book, Course, and Presentation (short and long) – samples Knowledge Worker Framework: Book, Course, and Presentation (short and long) – samples	http://www.wiscorp.com/EnterpriseDatabase.htm
Database Architecture Classes: sample	http://www.wiscorp.com/DatabaseDesign.htm
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
Metabase System (Free Version) Request form	http://www.wiscorp.com/freemb.html

The following documents are available for Whitemarsh Website Members. The URLs that follow provide descriptions of the pages. Members should log in and proceed to the appropriate page, e.g., Enterprise Database, find the book, paper, or course and perform the download.

Paper	URL
Data Management Program - Work Breakdown Structures	http://www.wiscorp.com/wwmembr/mbr_products_edb.html

Paper	URL
Enterprise Database Overview Enterprise Database Principles Information Systems Planning (book, course, and papers)	http://www.wiscorp.com/wwmembr/mbr_products_edb.html
Data Management Program - Metadata Architecture For Data Sharing Data Management Program - Database Interface Architectures Data Management Program - Projects And Data-Asset Product Specifications Data Management Program - Work Breakdown Structures Knowledge Worker Framework Database Objects Managing Database - Four Critical Factors	http://www.wiscorp.com/wwmembr/mbr_products_edb.html
Work Breakdown Structures Database Project Work plan Templates Information Systems Development Methodology Phases 1 and 2 Whitemarsh Project Estimating Work plan Development	http://www.wiscorp.com/wwmembr/mbr_products_dp.html