

Methodology and Technology Services



Web www.ies.aust.
 com

| [Home](#) | [Courses](#) | [Certification](#) | [Projects](#) | [Papers](#) | [EA Blog](#) | [Online Store](#) |
[Contact Us](#) |

[Home](#)
[Courses](#)
[Certification](#)
[Projects](#)
[Papers](#)
[TEN Archive](#)
[EA Blog](#)
[Contact Us](#)
[Search](#)
[Links](#)
[Online Store](#)

DEVELOPING A DATA WAREHOUSE

The Enterprise Engineering Approach

[Printable PDF Version](#)

By Alan Perkins, Managing Principal
© *Copyright 1995-1996 Visible Systems Corporation*

The business-driven, data-centric Information Engineering methodology provides an effective, productive, common sense approach to developing data warehouses. Information Engineering (IE) was pioneered by the Information Engineering Group of companies: [Information Engineering Services Pty Ltd](#) (Australia), [Visible System Corporation](#) (USA) and [Information Engineering Services Ltd](#) (New Zealand) - collectively called "Enterprise Engineering" in this paper.

Contents

- Paper "Developing a Data Warehouse"
 - [What is a Data Warehouse?](#)

- [Data Warehouse Benefits](#)
- [Data Warehouse Components](#)
- [Data Warehouse Structure](#)
- [Data Warehouse Development](#)
- [What is a Visible Advantage Encyclopedia](#)
- [What is the Enterprise Engineering Advantage?](#)
- [What is the Enterprise Engineering Approach](#)
- [Blueprint for a Data Warehouse](#)
- [The Enterprise Engineering Goal - Skills Transfer](#)
- [About the Author](#)
- [More Information](#)

WHAT IS A DATA WAREHOUSE?

A "Data Warehouse" is a collection of computer-based information that is critical to the successful execution of enterprise initiatives. IES

A data warehouse, which should more accurately be called an "information warehouse," is a subject-oriented database designed with enterprise-wide access in mind. It provides tools to satisfy the information needs of enterprise managers at all organizational levels - not just for complex data queries, but as a general facility for getting quick, accurate, and often insightful information. A data warehouse is designed so that users can recognize the information they want and access that information using simple tools.

A data warehouse is a blending of technologies, including relational and multi-dimensional databases, client/server architecture, graphical user interfaces and more. Data in a data warehouse differ from operational systems data in that they can only be read, not modified. Operational systems create, update and delete production data that "feed" the data warehouse. The principal reason for developing a data warehouse is to integrate operational data from various sources into a single and consistent warehouse that supports analysis and decision making within the enterprise.

For those enterprises that believe information is a valuable resource, a data warehouse is analogous to a physical warehouse. Operational systems create data "parts" that are loaded into the warehouse. Some of those parts are summarized into information "components" and stored in the warehouse. Data warehouse users make requests and are delivered "products" that are

created from the components and parts stored in the warehouse.

Data warehouse is one of the hottest industry trends - for good reason. A well defined and thought out data warehouse, properly implemented, can be a valuable competitive tool.

Back to [Contents](#).

Data Warehouse Benefits

- **More cost-effective decision making.** A data warehouse allows elimination of staff and computer resources required to support queries and reports against operational and production databases. This typically offers significant savings. Having a data warehouse also eliminates the resource drain on production systems when executing long-running, complex queries and reports.
- **Better enterprise intelligence.** Increased quality and flexibility of enterprise analysis arises from the multi-tiered data structures of a data warehouse that support data ranging from detailed transactional level to high-level summary. Guaranteed data accuracy and reliability result from ensuring that a data warehouse contains only "trusted" data.
- **Enhanced customer service.** An enterprise can maintain better customer relationships by correlating all customer data via a single data warehouse.
- **Business Process Re-Engineering.** Allowing unlimited analysis of enterprise information often provides insights into enterprise processes which may yield breakthrough ideas for re-engineering those processes. Just defining the requirements for a data warehouse results in better enterprise goals and measures.
- **Information System Re-Engineering.** A data warehouse that is based upon enterprise-wide data requirements, provides a cost-effective means of establishing both data standardization and operational system interoperability.

Back to [Contents](#).

Data Warehouse Components

The following primer describes each of the components of a data warehouse (see Figure). This description is based upon the work of W. H. Inmon, credited as the father of the data warehouse concept.

Current Detail.

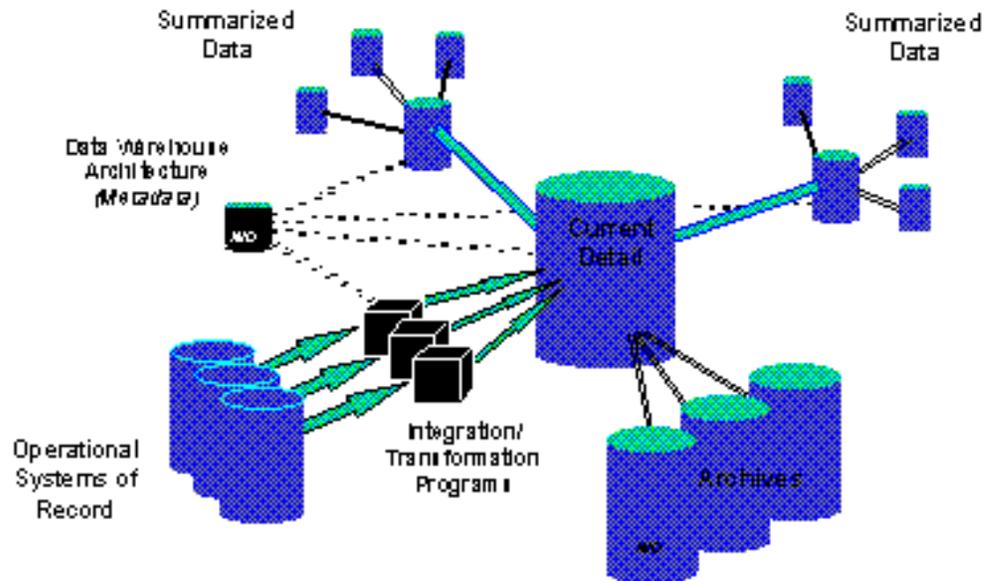
The heart of a data warehouse is its current detail. It is the place where the bulk of data resides. Current detail comes directly from operational systems and may be stored as raw data or as an aggregation of raw data. Current detail, organized by subject area, represents the entire enterprise, rather than a given application.

Current detail is the lowest level of data granularity in the data warehouse. Every data entity in current detail is a snapshot, at a moment in time, representing the instance when the data are accurate. Current detail is typically two to five years old. Current detail refreshment occurs as frequently as necessary to support enterprise requirements.

System of Record

A system of record is the source of the best or "rightest" data that feed the data warehouse. The "rightest" data are those which are most timely, complete, accurate, and have the best structural conformance to the data warehouse. Often the "rightest" data are closest to the source of entry into the production environment. In other cases, a system of record may be one containing already summarized data.

Data Warehouse Components



Integration and Transformation Programs

Even the "rightest" operational data cannot usually be copied, as is, into a data warehouse. Raw operational data are virtually unintelligible to most end users. Additionally, operational data seldom conform to the logical, subject-oriented structure of a data warehouse. Further, different operational systems represent data differently, use different codes for the same thing, squeeze multiple pieces of information into one field, and more. Operational data can also come from many different physical sources: old mainframe files, non-relational databases, indexed flat files, even proprietary tape and card-based systems. Thus operational data must be cleaned up, edited, and reformatted before being loaded into a data warehouse.

As operational data items pass from their systems of record to a data warehouse, integration and transformation programs convert them from application-specific data into enterprise data. These integration and transformation programs perform functions such as:

- Reformatting, recalculating, or modifying key structures
- Adding time elements
- Identifying default values
- Supplying logic to choose between multiple data sources
- Summarizing, tallying, and merging data from multiple sources

When either operational or data warehouse environments change, integration and transformation programs are modified to reflect that change.

Summarized Data

Lightly summarized data are the hallmark of a data warehouse. All enterprise elements (department, region, function, etc.) do not have the same information requirements, so effective data warehouse design provides for customized, lightly summarized data for every enterprise element (see Data Mart, below). An enterprise element may have access to both detailed and summarized data, but there will be much less than the total stored in current detail.

Highly summarized data are primarily for enterprise executives. Highly summarized data can come from either the lightly summarized data used by enterprise elements or from current detail. Data volume at this level is much less than other levels and represents an eclectic collection supporting a wide variety of needs and interests. In addition to access to highly summarized data, generally executives also have the capability of accessing increasing levels of detail through a "drill down" process.

Archives

Data warehouse archives contain old data (normally over two years old) of significant, continuing interest and value to the enterprise. There is usually a massive amount of data stored in the data warehouse archives that has a low incidence of access. Archive data are most often used for forecasting and trend analysis. Although archive data may be stored with the same level of granularity as current detail, it is more likely that archive data are aggregated as they are archived. Archives include not only old data (in raw or summarized form), they also include the metadata that describes the old data's characteristics.

Metadata

One of the most important parts of a data warehouse is its metadata - or data

about data. Also called data warehouse architecture, metadata is integral to all levels of the data warehouse, but exists and functions in a different dimension from other warehouse data. Metadata that is used by data warehouse developers to manage and control data warehouse creation and maintenance resides outside the data warehouse. Metadata for data warehouse users is part of the data warehouse itself and is available to control access and analysis of the data warehouse. To a data warehouse user, metadata is like a "card catalog" to the subjects contained in the data warehouse.

Back to [Contents](#).

Data Warehouse Structure

A data warehouse may have any of several structures:

- **Physical Data Warehouse** - physical database in which all the data for the data warehouse are stored, along with metadata and processing logic for scrubbing, organizing, packaging and processing the detail data.
- **Logical Data Warehouse** - also contains metadata including enterprise rules and processing logic for scrubbing, organizing, packaging and processing the data, but does not contain actual data. Instead it contains the information necessary to access the data wherever they reside.
- **Data Mart** - subset of an enterprise-wide data warehouse. Typically it supports an enterprise element (department, region, function, etc.). As part of an iterative data warehouse development process, an enterprise builds a series of physical data marts over time and links them via an enterprise-wide logical data warehouse or feeds them from a single physical warehouse.

Back to [Contents](#).

Data Warehouse Development

There are three popular "approaches" for data warehouse...two of which are bad.

1. "Data Dump" -- all enterprise data is replicated or made available with no attempt made to "scrub" or even categorize the data. Its like dumping all the contents in a physical warehouse in the middle of the floor and asking people to pick out what they need.
2. "Magic Window" to the data wherever it exists in the enterprise, again without ensuring data quality. It's like a big sack in which there are rubies and emeralds and gold nuggets and broken glass and rat droppings and poisonous snakes. At some point you will quit putting your hand in the bag.
3. "Strategic Data Warehouse" -- enterprise data based upon business requirements!

The Enterprise Engineering methodology and computer-based tools provide the flexibility and capability to easily develop an enterprise-wide, strategic data warehouse.

Back to [Contents](#).

What is a Visible Advantage Encyclopedia?

A Visible Advantage encyclopedia provides the mechanism and the medium for defining, storing and managing all data, process and management elements pertaining to an enterprise. It is at the heart of Visible Advantage tools and serves to connect the various lifecycle steps of the Enterprise Engineering methodology: Strategic/Tactical/Operational Planning, Data Modeling, Process Modeling, System Design and Development, Transition, and finally Production and Maintenance.

A typical encyclopedia includes:

- ***Enterprise Architecture:*** vision, mission, goals, strategies, critical success factors, policies, strengths, weaknesses, enterprise functions, enterprise measures, information needs, and their relationships.

Business Model: enterprise units, their hierarchy, and often key personnel identification.

Data Model: data entities (i.e., "persons, places, things") of importance in the enterprise, the properties of those entities, and their relationships to each other and to the enterprise architecture.

Process Model: how the business acts: business events and processes that define how the enterprise does what it does and how work flows through the enterprise; also how data entities are added, deleted, modified and used by the enterprise.

System Designs: the application, database, and information system designs that meet the information requirements of the enterprise.

A Visible Advantage encyclopedia is unique in that it allows every object of the database to be linked to any and every other object. This means that enterprise architecture (business planning) statements can be linked to data and process model objects as well as to system design components that support the requirements indicated by the planning statements. Having all related encyclopedia objects linked in this manner simplifies both encyclopedia maintenance and the integration of new objects. Because the encyclopedia is stored in an automated database, not only is every important object easily accessible, but any modifications made in one area are automatically rippled throughout the encyclopedia; thus, the enterprise model is always current.

Back to [Contents](#).

What is the Enterprise Engineering Advantage?

IES and visible have pioneered a business-driven, data-centric Enterprise Engineering methodology uniquely suited for data warehouse development. Coupled with Enterprise Engineering highly sophisticated, Computer-Assisted System Engineering (CASE) tool, Visible Advantage, the methodology provides a practical approach to data warehouse development.

Visible Advantage is a computer-based, business and Information Engineering support system that can be a valuable asset to any enterprise involved in data warehouse development. Visible Advantage includes an integrated encyclopedia (see box), extensive reporting capability, and state-of-the-art modeling, charting, analysis and information system design tools.

Visible Advantage includes the Enterprise Engineering -developed database

applications, reports, utilities, and interfaces. Visible Advantage has menus and forms that provide easy access to its powerful tools. These features, along with automatic prompts and on-line "Help," make Visible Advantage very userfriendly. Visible Advantage is Windows-based and is available in both stand-alone and network versions.

Back to [Contents](#).

What is the Enterprise Engineering Approach

IES and Visible have discovered that the key to success in data warehouse development is using an iterative approach that includes active participation of potential users.

Like any other large information systems project, data warehouse development can get bogged down if the scope is too broad and the number of people involved is too large. A clear purpose and scope is necessary to manage the application of information systems resources as well as manage the expectations of potential data warehouse users. Enterprise Engineering limits the scope by building a data warehouse a data mart at a time. Each data mart supports a single organizational element, enterprise function or business object (i.e. customer, product, account, etc.) and the scope of development is limited by the data mart requirements. For the initial data mart, which usually provides the data warehouse proof-of-concept, the scope must be sufficient to provide real, immediate, and high profile benefits. After the first data mart is developed and implemented, additional data marts can be developed and integrated over time as enterprise needs dictate and as resources are available.

Designing and developing a data warehouse using the Enterprise Engineering approach involves four very different activities: (1) identifying enterprise needs, (2) designing data warehouse architecture, (3) applying appropriate technology, (4) implementing the data warehouse.

- Identifying Enterprise Needs

Identifying enterprise needs is a major component in the life cycle for engineering any information system, and it is crucial when engineering a data warehouse. When developing operational systems, there is often one single enterprise management sponsor or one group of users with a clear view of

what they need, what the system should look like, and how it should function. Conversely, when developing a data warehouse, there are normally multiple potential users, each with a different idea of what a data warehouse is and what it will provide, and all requesting or demanding action. Because of this lack of focused direction, identifying precise enterprise needs is critical to the success of a data warehouse project.

Enterprise Engineering expresses enterprise data warehouse needs in terms of enterprise measures and critical success factors. An enterprise's business plans typically provide the basis for defining preliminary enterprise needs. The best Enterprise Engineering technique for refining these preliminary enterprise needs (or defining them if no business plan exists) is to conduct a series of facilitated focus group sessions. The participants in these sessions are the potential data warehouse users for whom the information in the data warehouse is important. Because the data warehouse is important to them, these individuals are willing to invest the time needed to describe their enterprise information needs.

- Building an Enterprise Model

Enterprise Engineering documents enterprise measures and critical success factors as planning statements in an Visible Advantage encyclopedia, and documents the supporting data entities in a corresponding data model. Data warehouse data entities are those whose values at any point in time are necessary to tell data warehouse users how well their enterprise is performing. Providing a clear and unambiguous definition of every key data entity, describing the way each is used, as well as defining derivation formulas, aggregation categories and time periods, are activities critical to capturing a clear understanding of an enterprise's measures. The resulting enterprise architecture model (see box), which links enterprise needs with data warehouse data entities and enterprise rules, becomes both requirements documentation and a source for communicating the contents of the data warehouse (its metadata) to its users.

- Determining Measurement Cycles

Completely defining an enterprise measure includes describing the cycles or time periods used for the measure. Are quarters, months or hours appropriate for capturing useful measurement data? How much historical data will be needed? This varies greatly by enterprise. The United States Federal Reserve Bank views enterprise measures in monthly, quarterly and annual increments and uses years of historical data to determine trends in the economy. A telephone sales operation, on the other hand, uses hourly

enterprise measures and may only keep a few weeks of information.

- Resolving Data Conflicts

A well-defined data warehouse model cannot contain homonyms, synonyms and other data definition conflicts. The reason these data conflicts may exist is because most enterprises have one or more major terms that are used by everyone in the enterprise but mean different things in different organizational units. One of the most commonly misused terms is "customer."

To the Accounting Department, "customer" could mean the organization (or individual) that receives a bill. "Customer" could also mean an individual receiving service or buying a product. To the Sales Department, "customer" could mean the organizations on whom the salesperson calls. Providing any one of these interpretations as the enterprise definition of customer would not meet the needs of the enterprise and would doom its data warehouse effort to failure due to "bad data."

Enterprise Engineering takes great pains to resolve all data conflicts in the data warehouse model before continuing with the next phase of the development cycle.

- Identifying Systems of Record

Clearly defining enterprise data warehouse architecture also involves identifying the correct source of raw operational data to populate the data warehouse. This effort also addresses possible integration and transformation logic. Identifying the systems of record for data warehouse data entities is one means of validating enterprise measures.

- Validating Measures

After identifying and defining enterprise needs, it is advantageous to communicate them throughout the enterprise. One of the best justifications for undertaking a data warehouse project is the synergy the enterprise achieves through the process of defining and then communicating its critical success factors and measures. Everyone in the enterprise becomes aware of precisely what defines success and how it is measured. In addition, the measures undergo a "reality check" by people who were not involved in their development. Their feedback is used for refining the measures.

Back to [Contents](#).

BLUEPRINT FOR A DATA WAREHOUSE

"Engineering" a data warehouse is a lot like "engineering" a physical warehouse. Both involve a rigorous development cycle and require the right tools. A building is constructed using architectural diagrams (blueprints) that clearly depict the building's infrastructure (structural elements, walls, electrical wiring, plumbing, etc.). Enterprise Engineering builds data warehouses from architectural models of enterprise infrastructure (policies, goals, measures, critical success factors, etc).

Blueprints are also used to enlarge a building or make any significant modifications. Without a diagram of the infrastructure such changes are quite difficult and very costly. It is the same with data warehouses. Enterprise Engineering first updates an enterprise's architecture model so that it reflects changes (new product lines or services, for example) and then modifies the data warehouse to support the changed enterprise.

Data warehouse engineering is easier less costly when based upon an accurate architectural model of the enterprise. Further, a data warehouse is easier to use and consistently produces desired outcomes when decision makers have access to an enterprise architecture (metadata) that accurately reflects enterprise infrastructure.

Designing Data Warehouse Architecture

After defining and thoroughly documenting enterprise needs (measures and critical success factors), Enterprise Engineering begins actual data warehouse architecture (metadata) design. This activity also involves active user participation in facilitated design sessions.

There are two types of data warehouse metadata: structural and access.

Structural metadata is used for creation and maintenance of the data warehouse. It fully describes data warehouse structure and content. The

basic building block of structural metadata is a model that describes its data entities, their characteristics, and how they are related to one another. The way potential data warehouse users currently use, or intend to use, enterprise measures provides insight into how to best serve them from the data warehouse; i.e., what data entities to include and how to aggregate detailed data entities. A Visible Advantage data warehouse data model provides a means of documenting and identifying both strategic and operational uses of enterprise measures. It also provides the capability to document multi-dimensional summarization of detail data.

Naturally, the number and specificity of data aggregation categories in a data warehouse will depend directly on the types of individuals who participate in design sessions.

- *Strategic thinkers* tend to be looking for the "big picture" answers and therefore need very few aggregation categories. The "roll-ups" for each strategic aggregation of data, however, can be quite complex.
- *Operational thinkers* have a tendency to want to dissect and review every measure by every category used in their part of the enterprise, and thus will tend to require large numbers of less complex aggregation categories.

Structural metadata identifies the system of record for all data warehouse data entities. It also fully describes the integration and transformation logic for moving each data warehouse entity from its system of record to the data warehouse. In addition, structural metadata defines the refreshment schedule and archive requirements for every data entity.

When the data warehouse structure changes, its metadata is changed accordingly. Old versions of the structural metadata are kept to document the changing nature of the data warehouse and allow access to archive data.

Structural metadata also includes performance metrics for programs and queries so that users and developers know how long programs and queries should run. Data warehouse performance tuning also uses these metrics.

Access metadata is the dynamic link between the data warehouse and end-user applications. It generally contains the enterprise measures supported by the data warehouse and a dictionary of standard terms including user-defined custom names and aliases. Access metadata also includes the location and description of data warehouse servers, databases, tables, detailed data, and summaries along with descriptions of original data sources and transformations.

Access metadata provides rules for drill up, drill down and views across enterprise dimensions and subject hierarchies, like products, markets, and customers. Access metadata also allows rules for user-defined custom calculations and queries to be included. In addition, access metadata contains individual, work group, and enterprise security for viewing, changing, and distributing custom calculations, summaries or other analyses.

Applying the Correct Technology Solution

Only after fully defining enterprise requirements and designing the data warehouse architecture, should an enterprise begin to select the technology for the data warehouse. Key technology issues, in addition to determining the hardware/software platform for the data warehouse, include developing programs for loading information into the data warehouse, implementing access control (security) mechanisms and selecting one or more user interface tool sets.

- Hardware/Software Platforms

The following are some important considerations for determining a hardware platform:

- How much data will be in the data warehouse and how much can the platform economically accommodate? How scalable is the platform? Is it optimized for data warehouse performance? Will the platform support the software selected for the data warehouse?
- Concurrent with hardware selection is the selection of system software to support the data warehouse. Among the choices are operating systems, development software, and database management systems. The structure and size of the data warehouse will determine system software requirements. For example, a data warehouse that includes data marts will require not only relational technology, but multidimensional access and a client/server architecture.
- Integration and Transformation Programs

Integration and transformation programs are necessary to extract information from operational systems and databases for both initial load and subsequent updates of the data warehouse. Sometimes, it is possible to develop a single program for both initial load and periodic updates of the data warehouse, but often circumstances make this an unacceptable development option.

A separate initial load program is necessary when the volume of initial data is so large that it cannot be transferred without adversely impacting other users of the operational systems. This is particularly true when initial load and update volumes are significantly different.

Separate programs should also be considered for capturing historical data from the operational systems for loading into the data warehouse. Usually this is a one-time process. An additional complication involves historical data maintained separately from the operational systems (many operational systems only maintain the most recent values for data). This situation usually requires retrieval of historical data from archive and backup files.

Under either of these circumstances, one set of integration and transformation programs initially loads the data warehouse and a second set periodically updates the data warehouse. Update programs are generally smaller and simpler than programs developed to load the data warehouse. Update programs often are built into operational systems to trap new occurrences of data as they are added. This works best for well-documented, in-house operational systems. Update programs that extract data from commercial off-the-shelf software or from older, poorly documented, legacy systems typically capture just the changes made since the last update.

Security

A data warehouse is a read-only source of enterprise information; therefore developers need not be concerned with controlling create, update and delete capabilities. But, developers will need to address the trade off between protecting a valuable corporate asset against unauthorized access and making the data accessible to anyone within the enterprise who can put it to good use. The best solution IES and Visible have found is to allow everyone in the enterprise to have access to the enterprise measure definitions and derivations, but only allow access to the underlying detailed data on an approved, need-to-know basis.

User Interfaces

Data Warehouse users get useful information from the data warehouse through user interfaces. It is these user interfaces that have the most impact on how effective and useful the data warehouse will be perceived by its users. Two criteria for selecting an effective user interface are ease of use and performance. For ease of use, most enterprises turn to graphical user interfaces. For performance, developers must ensure that the hardware/

software platform fully supports and is optimized for every chosen user interface.

The most important selection criteria for user interfaces are the information needs and the level of computer literacy of potential users who will retrieve the information they need from the data warehouse. The following data warehouse user categories are based on levels of literacy and information needs.

- *Information Systems Challenged* - data warehouse users who are hopelessly lost when it comes to information systems. In management roles they rely on their secretaries or assistants to retrieve information for them. These users need an extremely easy to use and highly graphical interface.
- *Variance Oriented* - users who are focused on the variances in numbers over time. These users mainly want a set of standard reports that they can generate or receive periodically so that they can perform their analyses.
- *Number Crunchers* - users who are spreadsheet aficionados. They will take whatever data are available and refine it, re-categorize it and derive their own numbers for analyzing and managing the enterprise. Their needs can best be met by providing a spreadsheet extract output format for any reports or ad hoc queries provided.
- *Technically Oriented* - users who are either already familiar with computers or have sufficient motivation to learn and use everything they can get their hands on. These people want to have complete control over the way they retrieve and format information. They are often business or systems analysts who have moved into an enterprise function. They want to have all of the tools the data warehouse development staff uses.

Most enterprises have all of these categories of individuals. This makes it advisable to provide each type of data warehouse user interfaces.

The final user interface criterion is that it support the access metadata designed for the data warehouse. If a user interface is easy to use, allows all potential users to get the information they need in the format they need, and does it in an acceptable amount of time, it is the right interface.

Implementing the Data Warehouse

Data warehouse implementation includes loading the "rightest" data, designing a user interface "look and feel," developing standard queries and

reports, and thoroughly training data warehouse users. In the Enterprise Engineering approach, the enterprise is totally responsible for ensuring that their data warehouse is implemented and becomes an enterprise asset.

Back to [Contents](#).

The Enterprise Engineering Goal - Skills Transfer

Although "data warehouse" is a relatively new term, many Executive Information Systems (EIS), Decision Support Systems (DSS) and Management Information Systems (MIS) were developed by IES, Visible and their clients using the underlying concept long before the term existed. Enterprise Engineering perfected its approach to data warehouse development through years of experience. Recently, IES and Visible consultants successfully assisted the Securities and Exchange Commission and Blue Cross/Blue Shield of the National Capital Area with their data warehouse development efforts.

The Enterprise Engineering methodology and tools are uniquely suited to support development of an enterprise data warehouse. Visible Advantage is the only integrated CASE tool that allows enterprise needs and measures to be linked directly to the data warehouse data model, data dictionary, integration and transformation process models, and the data warehouse database design in a single relational architecture. The data warehouse architecture becomes the data warehouse structural metadata and is the source for access metadata.

Just as a powerful word processing system is not very useful to someone who does not write, Visible Advantage is not useful without appropriate knowledge and skills in the acquisition, interpretation and representation of all the details that go together to make up a model of an enterprise and develop a data warehouse architecture.

Enterprise Engineering helps clients gain the necessary expertise to effectively use Visible Advantage for data warehouse development through facilitation, training, education and consulting. IES and Visible pride themselves on their ability to transfer skills and knowledge that allow clients to gain mastery of the Enterprise Engineering methodology and tools. Generally this requires three complete development cycles (defining and modeling enterprise needs, designing the data warehouse architecture,

applying appropriate technology, implementing a data mart).

1. During the first cycle, one or more IES or Visible consultants help the enterprise complete a prototype data mart for the data warehouse while training enterprise personnel on a data warehouse "Action Team." IES and Visible consultants are actively involved in every aspect of this initial development cycle. During this and every phase of data warehouse development, enterprise analysts, developers and data warehouse users receive appropriate training that ensures they have the skills and knowledge to effectively participate. This "just-in-time" training is a hallmark of IES and Visible.
2. In the second cycle, IES and Visible consultants closely monitor and coach enterprise personnel as the Action Team completes the next data warehouse element.
3. Finally, internal enterprise personnel perform a complete data warehouse development cycle with minimal assistance from IES and Visible, typically in the form of progress and quality assurance reviews. By the end of the third cycle, enterprise personnel are fully capable of developing data warehouse components on their own.

Information is a valuable resource. A well-defined data warehouse, properly implemented, can be a valuable tool for managing and using that resource. It translates the vast volumes of detailed unorganized data an enterprise captures via its operational systems into useful feedback, predictors and warnings that help data warehouse users at every organizational level make more informed decisions.

The Enterprise Engineering approach to data warehouse development results in a useable, effective information management tool that exactly meets the needs of an enterprise, business or government, large or small.

Back to [Contents](#).

Contact Details

For more information on Data Warehouse tools and techniques, please contact:

North America

Visible Systems Corporation
201 Spring Street Lexington MA 02421 USA
Phone: +1-781-778-0200 · Fax +1-781-778-0208
Web Site: <http://www.visible.com>
Email: mcesino@visible.com

Asia-Pacific

Clive Finkelstein, Managing Director
Information Engineering Services Pty Ltd
PO Box 246, Hillarys Perth WA 6923 Australia
Phone: +61-8-9402-8300 Fax: +61-8-9402-8322
Web Site: <http://www.ies.aust.com/>
Email: cfink@ies.aust.com

Back to [Contents](#).

| [Home](#) | [Courses](#) | [Certification](#) | [Projects](#) | [Papers](#) | [TEN Archive](#) | [EA Blog](#) | [Online Store](#) | [Contact Us](#) | [\[Search\]](#) |

(c) Copyright 2004-2006 Information Engineering Services Pty Ltd. All Rights Reserved.