

Methodology and Technology Services



www.ies.aust.
Web com

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

BUSINESS RE-ENGINEERING AND THE INTERNET: Transforming Business for a Connected World

[Printable PDF Version](#)

By Clive Finkelstein, Managing Director
Copyright © 1993-1998 Information Engineering Services Pty Ltd.

Contents

■ Business Re-Engineering and the Internet

- [Abstract](#)
- [The Problems of Change](#)
- [Manual Processes and Systems](#)
- [Automated Process and Systems](#)
- [The First Era of the Information Age](#)
- [The Second Era of the Information Age](#)
- [Business Re-Engineering for the Second Era](#)
 - [Business Re-Engineering from Business Plans](#)
 - [Business Re-Engineering from Business Processes](#)
 - [Business Re-Engineering from Business Information](#)
 - [Identifying Business Processes from Data Models](#)
 - [Identifying Cross-Functional Processes from Data Models](#)
 - [Re-Engineering Opportunity Analysis](#)
- [The Status of Internet and Intranet Technologies](#)
 - [Box 1: A Status Update of Internet and Intranet Technologies](#)
- [Business Re-Engineering and the Internet](#)
- [Conclusion](#)
- [References and Glossary](#)
- [The Author: Clive Finkelstein](#)
- [More Information](#)

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

ABSTRACT

The Internet and corporate Intranets present opportunities to re-engineer business processes for direct access between customers and suppliers. Re-engineering for this environment requires close integration of business plans, business processes and business information, to ensure that systems are built that are closely aligned with strategic directions. A new generation of I-CASE tools is also emerging that can automatically analyse data models to identify cross-functional processes. These present re-engineering opportunities that benefit from the emerging open architecture environment of the Internet and Intranets.

Back to [Contents](#).

THE PROBLEMS OF CHANGE

The Internet and its corporate counterpart, the Intranet, are transforming the competitive landscape with a rapidity and in ways never thought possible ([Finkelstein, 1996](#)). Organizations are faced with unprecedented competition on a global basis. To compete effectively, they must change: those that fail to see the need for change will not survive. This change, in most cases, involves re-engineering the business. This paper shows how a focus on business re-engineering, and development of information systems that directly support strategic business plans, allows managers to take control and turn tomorrow's chaotic environment to their competitive advantage by using the Internet and Intranets.

To succeed in today's environment an organization must focus first on customers and markets, rather than on products and production. To compete effectively in today's market environment, flexibility is key: time to market has to go to virtual zero ([Zachman, 1992](#)). This suggests a strategy of **Assemble-to-Order**: of products, custom-built from standard components, and tailored to each customer's specific needs.

An assemble-to-order strategy applies not only to manufacturing and assembly, but to most service industries as well: such as custom-tailored banking or insurance products, or government services. It also applies to systems development. The solutions are well known. They involve the integration of business and IT: integration on the business side using strategic business planning and business re-engineering; and integration for IT using client-server systems and object-oriented development. The Internet and Intranets also assist.

We will first discuss well-known problems associated with manual and automated systems. We will then see how data models can help us identify re-engineering opportunities for business processes. We will discuss how open architecture environments established by Internet and Intranet technology can be utilised to re-engineer processes in ways that were difficult to achieve before.

Back to [Contents](#).

MANUAL PROCESSES AND SYSTEMS

Consider a typical business that accepts orders for products or services from their Sales Dept, placed by customers. These orders are processed first by the Order Entry Dept, and then by the Credit Dept. Each department needs details of the customer, such as name and address, and account balance, and details of the order. These are saved in customer files and order files, held and maintained by each department as shown in Figure 1.

In satisfying these orders, items used from inventory must eventually be replaced. These are ordered by the Purchasing Dept from suppliers, and are then later paid by the Accounts Payable section of the Accounts

Dept. Details of name and address, and the account balance due to the supplier, are also saved in supplier files and creditor files. These are held redundantly and are maintained separately by each department as also shown in Figure 1.

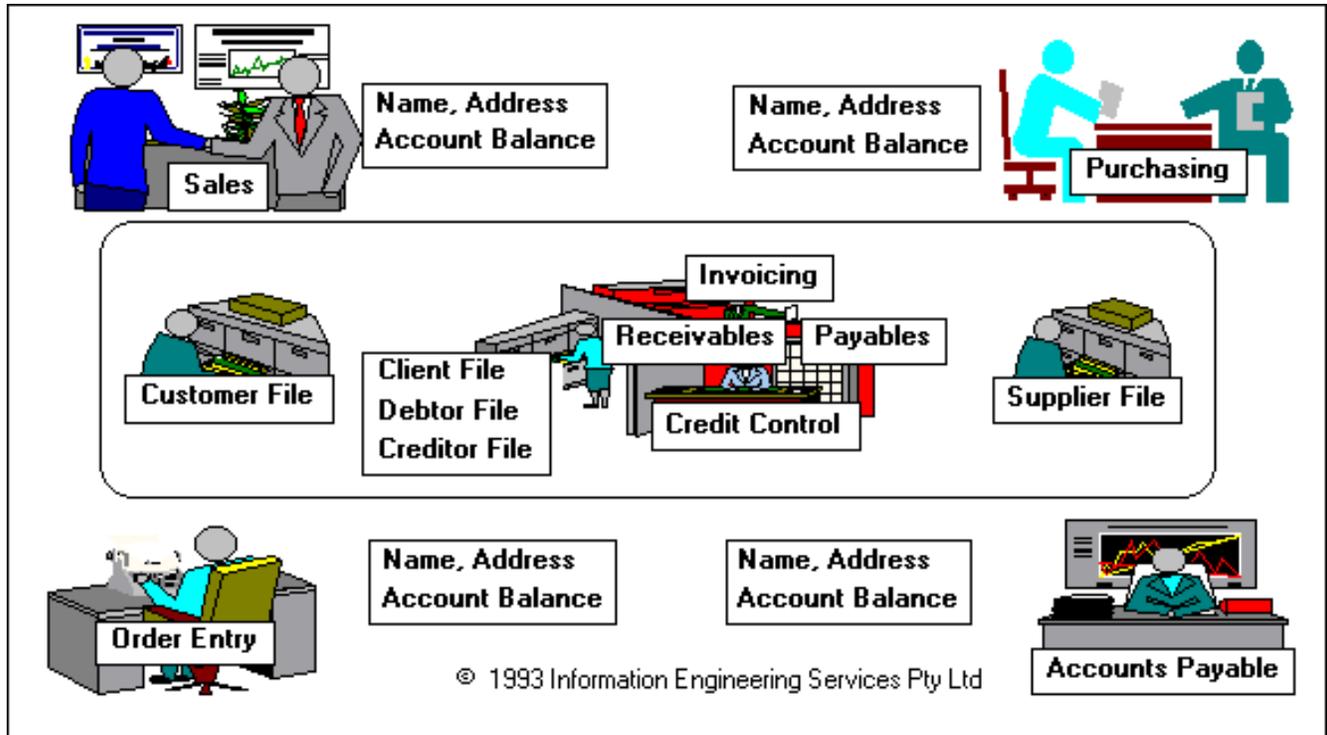


Figure 1: The same data often exists redundantly in an organization. Each redundant data version must be maintained current, and up-to-date. This invariably leads to the evolution of redundant processes.

To ensure these redundant data versions are kept current and up-to-date and all versions consistently refer to the same information, any change to one data version must be also made to all other affected versions. For example, a customer notifies Sales of a change of address. That address change must be made not only in the Sales Dept customer file, but also in the customer file maintained by the Order Entry Dept. The same customer details are held and maintained redundantly by the Credit Dept in their client file, and by the Invoicing Section of the Accounts Dept in their debtor file.

And if the customer also does business with the organization as a supplier, that change of address must be made to the supplier file maintained by Purchasing and to the creditor file maintained by Accounts Payable in Accounts. The address change must be made to **every** redundant data version, so that all information about the customer is correct and is consistent across the organization.

This is not a computer problem: it is an organization problem! But its resolution over the years has defined the manual processes and the organization structures adopted by countless organizations. In our earlier example, processes are defined to capture the details of customers and make changes to those details when needed later, as in the change of customer address. Many of these processes are also redundant, but are necessary to maintain the data versions all up-to-date. Redundant data thus leads to redundant processes.

And of course, people have been allocated to carry out these redundant processes in each department. This staffing is unproductive and adds nothing to the bottom line: in fact, it reduces profitability. It introduces unnecessary delays in serving the customer, so reducing customer service - leading often to competitive disadvantage.

Back to [Contents](#).

AUTOMATED PROCESSES AND SYSTEMS

In the 1980s office automation was introduced to solve the problem. But it focused on the symptom: the time that the Address Change Form took to reach each affected department. It did not address the true problem, the redundant data - it only sped up the paper! To resolve the problem common data (name and address in our example) should be stored once only, so all areas of the business that are authorized to access it can share the same data version. This is illustrated in Figure 2.

While name and address is common to many areas, there is also data needed only by certain areas. An organization may have more than one role in its business dealings, shown by Organization Role in Figure 2: one role may be as a customer and another role as a supplier. For example, the Credit Dept must ensure that the value of the current order, when added to the outstanding debtor balance still to be paid (maintained by Accounting) does not exceed the customer's credit limit. And Accounting must be aware of the creditor balances that are due to be paid to organizations that we deal with as suppliers. While this data is specific to each of these areas, an organization's name and address is common - regardless of its role as a customer, or as a supplier - and so is shared by all, as shown in Figure 2.

It is in the identification of common data that the greatest impact can be made on the efficiency and responsiveness of organizations to change. Once applied, changes made to common data are available immediately to all areas that need the data to be current. And because common data is stored only once, redundant processes that maintained redundant data are no longer needed. Business processes that evolved over many years to ensure all areas had access to needed data (by each department maintaining its own data version) now share common data and are immediately aware of any changes.

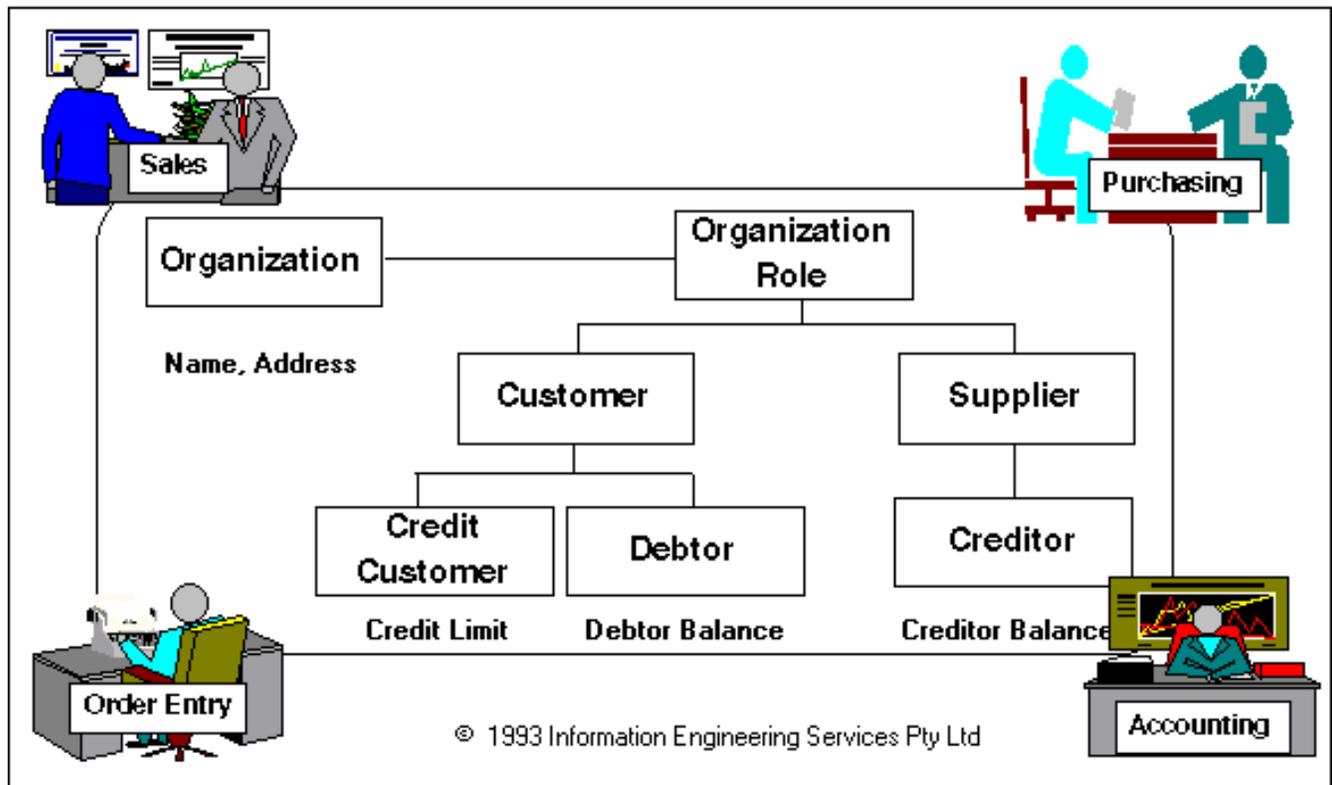


Figure 2: Common data can be shared as one data version, and made available to all authorized to use it. Specific data is then managed by each area that needs it.

THE FIRST ERA OF THE INFORMATION AGE

The way in which an organization structures itself today, when common data is readily available and can be easily shared is quite different from the way it had to be organized when that data was difficult to obtain. New business processes emerge that often cross previous functional boundaries. This leads to *Business Re-Engineering* (BRE) The strong interest in Business Process Reengineering (BPR) has addressed only a subset of the broader subject of business re-engineering. Organizations which approach BPR without recognizing and first correcting the redundancy problems of organizational evolution discussed above, are inviting trouble. This is discussed further, shortly.

We are now seeing the emergence of the "New Corporation", defined by Tapscott and Caston (1992) as *"the fundamental restructuring and transformation of enterprises, which in turn cascades across enterprises to create the extended enterprise and the recasting of relationships between suppliers, customers, affinity groups and competitors."* In their book (Tapscott et al, 1993) they discuss that organizations today are now moving from the First Era of the Information Age, to the Second Era.

They categorize the First Era by the traditional system islands built for established organization structures, using computers for the management and control of physical assets and facilities, for the management and support of the human resource, and for financial management and control systems. Computers have automated the existing processes used by an organization, replicating redundant data and manual processes with redundant files or data bases, and redundant computer processing. But these automated systems are far more resistant and difficult to change than the manual systems they replace. Organizations have buried themselves in computer systems that have now set like concrete: the computer systems introduced to improve organizational responsiveness now inhibit the very business changes needed to survive!

One of the original roles of middle managers was to implement internal procedures and systems that reflected directions and controls set by senior management. Their other role was to extract specific information, needed by senior management for decision-making, from underlying operational data at all levels of an organization. Organizations that downsized by only eliminating middle managers are now vulnerable. But those that downsized also by eliminating redundant data as well as redundant processes have enjoyed dramatic cost reductions - while also ensuring that accurate information is made available for management decision-making.

It is true that with accurate, up-to-date information available electronically and instantly, many layers of management in the First Era are no longer needed. But if this accurate information is not available, organizations invite disaster if they do not first implement effective systems to provide that information before removing middle managers. Only when the earlier problems of redundant data and redundant processes are resolved can downsizing truly be effective. Then decision-making can be faster, when access to accurate, up-to-date information is available corporate-wide.

Back to [Contents](#).

THE SECOND ERA OF THE INFORMATION AGE

Tapscott and Caston next described the Second Era of the Information Age as that which supports open, networked enterprises. These new corporations move beyond constraints of organizational hierarchy. In this Second Era, re-engineered processes and systems in the new corporation not only cross previous functional boundaries. They move beyond the organizational hierarchy - utilising computer systems that can directly link suppliers and customers. For example, insurance companies link directly with agents who sell their products: insurance policies that are uniquely tailored to satisfy their customers' exact insurance needs. With the Internet today, we are now moving into the Second Era.

Similarly airlines link with travel agents and convention organizers. The services they offer are not only booked airline flights, but also accommodation, car rental, and business meetings or holidays tailored uniquely to each customer's needs. In addition: banks provide direct online account access for their customers; manufacturers link to terminals in the trucks of their suppliers for just-in-time delivery; and

governments provide information kiosks for public access, so that people can obtain the specific information they need. These are all examples of business re-engineering, in the spirit so memorably encapsulated in the title of the landmark paper by Michael Hammer (1990): "*Reengineering Work: Don't Automate, Obliterate*".

The important point to recognize with *Business Re-Engineering*, and its subset of *Business Process Reengineering*, is their vital link with the development of computer application systems. But systems development has seen dramatic change. Integrated Computer-Aided Software Engineering (I-CASE) tools and methodologies are available that automate most manual, error-prone tasks of traditional systems development. These result in the rapid development of high quality systems that can be changed easily, and fast, resolving many of the problems discussed earlier.

An understanding of computer technology was a prerequisite of the traditional systems development methods of the First Era, and it was hard for business managers to participate effectively. The new generation of I-CASE tools for the Second Era achieve dramatic success by harnessing the knowledge of business experts. This knowledge is held by business managers and their staff, not computer analysts. When business experts use *business-driven Information Engineering* ([Business-Driven IE](#)) in a design partnership with computer experts, redundant data and processes are readily identified.

Using the knowledge of business experts across different areas that share common data, integrated data bases that eliminate redundant data and redundant processes are defined using business-driven CASE tools ([IE CASE](#)). Automatic analysis of common data by these CASE tools identifies new business processes that cross functional boundaries, and where appropriate also cross enterprise boundaries as discussed later. These cross-functional processes are automatically identified by the software and suggest common, shared processes. These can be identified and implemented across an organization, for improved efficiency and effectiveness. Re-engineering opportunities thus emerge.

Data bases and systems are designed which are of high business quality, and able to be implemented by computer experts using the most appropriate computer technology for the business: decentralized in a client-server environment, or instead centralized. Data bases can be automatically generated for any SQL-dialect RDBMS, such as IBM DB2, Oracle, CA/Open-Ingres, Sybase, Microsoft SQL Server and other RDBMS products. These systems can be built using a wide variety of computer languages or development tools, as object-oriented systems that share common data and common logic. This enables systems to be built rapidly and changed easily. The I-CASE tools discussed earlier automatically derive object-oriented logic from integrated data identified by the business experts.

The result is a dramatic gain in systems development and maintenance productivity and quality. Systems can now be built rapidly to meet the changing business requirements imposed by the intense competitive environment of today. These systems become competitive weapons that achieve success: not just by eliminating redundant data and processes, and duplicated staffing - so leading to huge cost savings. They also provide management with accurate, up-to-date, consistent information that was difficult, if not impossible, to obtain with confidence before. In this way, IT achieves its true potential, as corporations move to the Second Era of the Information Age.

The rest of this paper illustrates how this is achieved, using the Internet and Intranet for deployment.

Back to [Contents](#).

BUSINESS RE-ENGINEERING FOR THE SECOND ERA

As discussed above, we need to consider not only redundant data versions, but also the redundant processes which have evolved to keep redundant data versions up-to-date. Integrated data models not only eliminate redundant data versions, but also eliminate redundant processes. Data and processes represent *Business Information* and *Business Processes* which both must support *Business Plans* defined by management, as

shown in Figure 3.

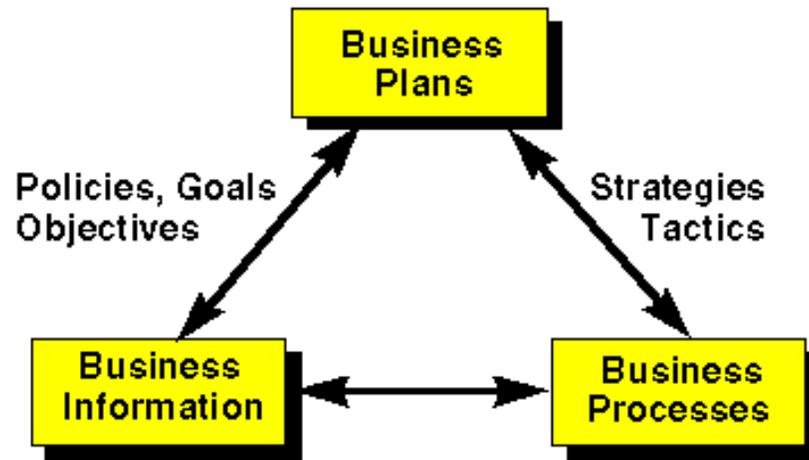


Figure 3: Business Re-Engineering improves all three areas essential to business effectiveness: Business Plans, Business Processes and Business Information.

Business Re-Engineering from Business Plans

Business plans represent the ideal starting point for Business Re-Engineering, as they apply at all management levels. When defined explicitly by senior managers they are called strategic plans; when defined by middle managers they are called business plans. At lower management levels they are defined implicitly as part of the manager's job description. We will use the generic term Business Plan to refer to all of these.

Business plans define the directions set by management for each business area or organization unit. They indicate the mission of the area and its defined policies, critical success factors, its goals, objectives, strategies and tactics. They are catalysts for definition of business processes, business events and business information as follows.

Policies are qualitative guidelines that define boundaries of responsibility. In a data model, policies are represented by related groups of data entities ([Data Entity](#)). The data entities defined within a business area thus together represent the policies that apply to the management and operation of that area. A policy also establishes boundaries for business processes that are carried out by one business area, and processes that relate to other business areas. An example of a policy is:

-
- **Employment Policy:** We only hire qualified employees.
-

Critical Success Factors (CSFs) - also called *Key Performance Indicators* (KPIs) or *Key Result Areas* (KRAs) - define a focus or emphasis that achieves a desired result. They lead to the definition of goals and objectives. *Goals* and *objectives* are quantitative targets for achievement. Goals are long-term; objectives are short-term. They indicate *WHAT* is to be achieved, and have quantitative characteristics of *measure*, *level* and *time*. The *measure* is represented by data attribute(s) within entities ([Data Attributes](#)). The *level* is the value to be achieved within an indicated *time* for the goal or objective. These attributes represent management information, and are generally derived from detailed attributes by processes at the operational level. They provide information that managers need for decision-making. An example of a measurable objective is:

- **Hiring Objective:** To be eligible, an applicant must exceed a score of 70% on our Skills Assessment Test at the first attempt.
-

There are many alternative *strategies* that managers may use to obtain the information they need. Strategies may contain more detailed *tactics*. Strategies and tactics define *HOW* the information is provided, and *HOW* it is derived. Together they lead to the definition of business processes. Examples of a strategy and business process are:

- **Assessment Strategy:** Interview each applicant and match to position criteria, then administer the relevant Skills Assessment Test.
 - **Evaluation Process:**
 1. Review the completed Application Form to ensure all questions have been satisfactorily completed.
 2. Check that required references have been provided.
 3. Select and administer relevant Skills Assessment Test.
 4. Note total time taken to complete all questions.
 5. Mark responses and calculate overall score.
 6. Write score and completion time on Application Form.
-

A strategy is initiated by a *Business Event*, which in turn invokes a business process. Without a business event, nothing happens: business plans, policies, goals, objectives, strategies and processes are merely passive statements. A business event initiates a business activity or activities - ie. business process(es). A business event example is:

- **Interview Event:** Schedule an interview date with the applicant.
-

Documented planning statements of mission, critical success factors, policies, goals, objectives, strategies, tactics and business events are allocated to the business area(s) or organization unit(s) involved in, or responsible for, those statements: in a *Statement - Business Area Matrix* or *Statement - Organization Unit Matrix*. This enables the subset of planning statements for each area or unit to be clearly identified. Strategic plans that define future directions to be taken by an organization represent the most effective starting point for Business Re-Engineering. But in many organizations today the plans are obsolete, incomplete, or worse, non-existent. In these cases, another apex of the triangle in Figure 3 can be used as the starting point: either business processes, or business information.

Business Re-Engineering from Business Processes

Existing business processes, reviewed and documented by narrative description and/or Data Flow Diagrams (DFDs) show how each process is carried out today. Business areas or organization units that are involved in, or responsible for, a process are identified and documented in a *Process - Business Area Matrix* or a *Process - Organization Unit Matrix*.

A business event is the essential link between a business plan and a business process. In the plan, an event is defined as a narrative statement. It can be a physical transaction that invokes a business process. Or it may represent a change of state. The strategy or tactic that is initiated by an event is documented in an *Event - Strategy Matrix*. This link must be clearly shown. The process invoked by each event should also be clearly indicated: documented in an *Event - Process Matrix*.

Without links to the plan, the business reason(s) why the process exists is not clear. It may be carried out only because *iwe have always done it that way.*" If the process cannot be seen to support or implement part of the plan, or provide information needed for decision-making, then it has no reason to remain. As the past fades into history, it too can be discarded as a relic of another time. To re-engineer these processes without first determining whether they are relevant also for the future is an exercise in futility. Worse still, the danger is that management feel they have done the right thing ... when they may have only moved around the deckchairs on their own "Titanic".

If the process is essential, then the strategies implemented by the process must be clearly defined. Associated goals or objectives must be quantified for those strategies. Relevant policies that define boundaries of responsibility for the process and its planning statements must be clarified. Missing components of the plan are thus completed, with clear management direction for the process.

The third apex of Figure 3 is an alternative starting point for Business Re-Engineering. In fact, business information is a far more powerful catalyst for re-engineering than business processes, as we will see.

Business Re-Engineering from Business Information

Data models developed for business areas or organization units should ideally be based on directions set by management for the future. These are defined in business plans. Where business plans are not available, or are out-of-date, or the reasons why business processes exist today are lost in the dark recesses of history, data models of business information provide clear insight into future needs.

Data models can be developed from any statement, whether it be a narrative description of a process, or a statement of a policy, goal, objective or strategy. The redundant data versions that have evolved over time (see Figure 1) are represented as data models, consolidated into integrated data models. Data versions from different business areas are integrated so that any common data can be shared by all areas that need access to it. Regardless of whichever area updates the common data, that updated data is then available to all other areas that are authorized to see it.

With this integration, redundant business processes - earlier needed so that redundant data versions are maintained up-to-date - are no longer required. Instead, new processes are needed. As common data is integrated across parts of the business, data that previously flowed to keep the redundant data versions up-to-date no longer flows. With integrated data models, implemented as integrated data bases, data still flows to and from the outside world - but little data flows inside the organization. The original processes that assumed data existed redundantly may no longer work in an integrated environment. New, integrated, cross-functional processes are required. But how can cross-functional processes be identified? Data Flow Diagrams provide little guidance in this situation. And *Affinity Analysis* provides little help either. This technique is used by the earlier DP-driven variant of IE to group related entities into business areas or subject areas ([Data Model Analysis](#)). But it is highly subjective. It depends on the knowledge that the data modeler has of the business; of data modeling; and the thresholds that are set. As a technique, it is not repeatable. It is potentially dangerous, as indicated next.

Where allowance is made for its subjectivity, affinity analysis can be still be useful. But when its results are accepted blindly, without question, the end result can be disaster. It lacks rigor and objectivity. It can lead to the grouping of more data in a subject area than is needed to deliver priority systems. This will require more resources to develop those systems; they will take longer and cost more. This is merely embarrassing, as in: *"the IT department has overrun its budget yet again."*

But the real danger is that essential, related data may not be included in the subject area. This related data

may indicate inter-dependent processes that are essential for the correct functioning of the processes represented by the priority systems. The end result? When delivered, the systems may not support all business rules that are essential for correct functioning of the business process. The systems may be useless: developed at great cost, but unable to be used. This situation is not embarrassing: it is disastrous! At best, it represents wasted development time and cost. At worst, it can affect an organization's ability to change rapidly, to survive in today's competitive climate.

Related data that indicates existence of inter-dependent processes leads to the definition of cross-functional processes, derived from data models. These suggest re-engineered business processes.

Identifying Business Processes from Data Models

Business processes can be identified from an analysis of a data model, based on the concepts of *Entity Dependency*. This is an objective technique, described in my recent book ([Finkelstein, 1992](#)). Its importance for Data Administrators was acknowledged in Database Newsletter ([McClure, 1993](#)). Entity dependency is rigorous and repeatable: it can be applied manually, or can be fully automated. When used to analyse a specific data model, the same result will *always* be obtained - regardless of whether the analysis is done by man or machine.

Entity dependency automatically identifies all of the data entities that a specific process is dependent upon, for referential integrity or data integrity reasons. It will automatically identify inter-dependent processes, and indicate cross-functional processes. It uncovers and provides insight into business re-engineering opportunities.

Consider the following example, based on the analysis of a data model developed for the business processes discussed earlier in this paper: involved in Sales and Distribution. Figure 4 shows an integrated data model that consolidates the previously separate functions of Order Entry, Purchasing, Product Development and Marketing. This data model represents business processes in each of these business areas, stated as follows:

- **Order Processing:**

"A customer may have many orders. Each order must comprise at least one ordered product. A product may be requested by many orders."

- **Purchase Order Processing:**

"Every product has at least one supplier. A supplier will provide us with many products."

- **Product Development:**

"We only develop products that address at least one need that we are in business to satisfy."

- **Marketing:**

"We must know at least one or many needs of each of our customers ."

Figure 4 will be used to illustrate an important principle of business-driven Information Engineering, used to identify process re-engineering opportunities from a data model. This principle is stated as:

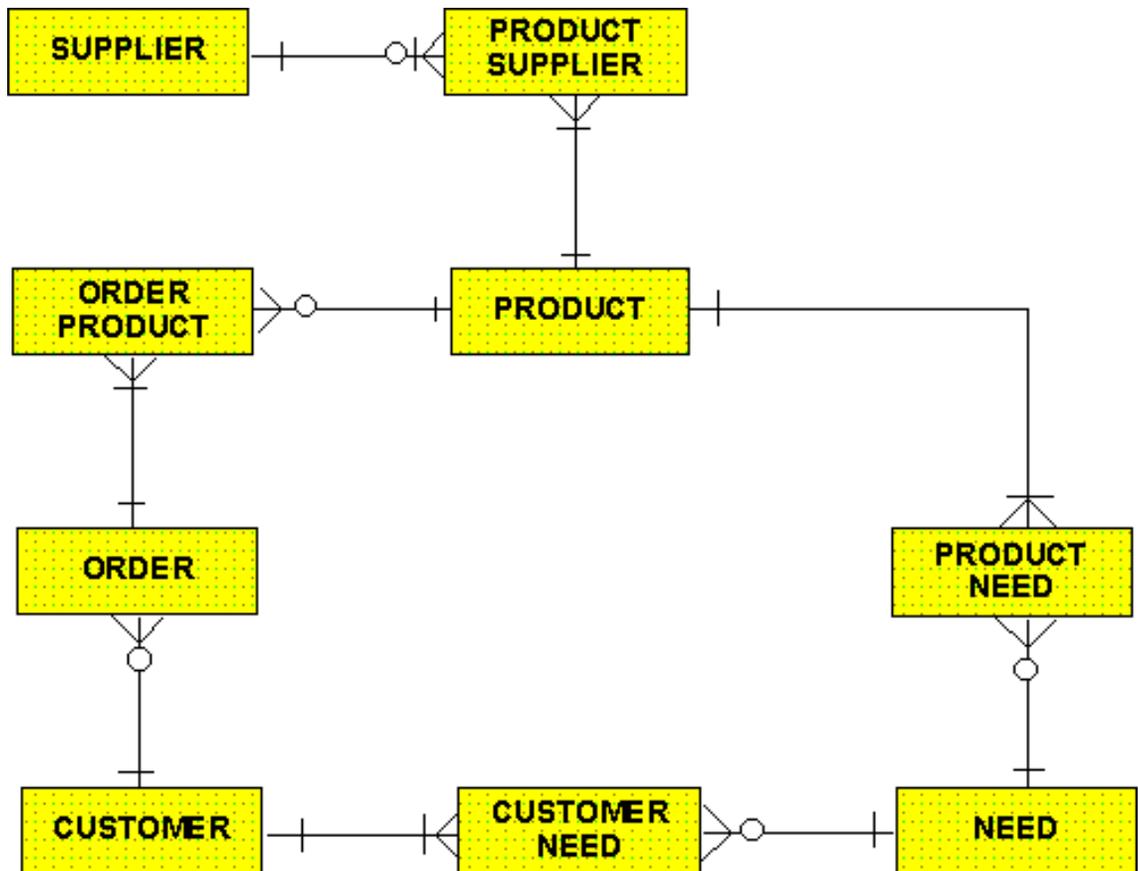
Intersecting entities in a data model represent functions, processes and/or systems.

This leads to identification of cross-functional processes that arise from integration of the Order Entry, Purchasing, Product Development and Marketing functions as we shall see.

Referring to Figure 4, ORDER PRODUCT is an intersecting (or "associative") entity formed by decomposing the many to many association between ORDER and PRODUCT. It represents the *Order Entry Process* used in the Order Entry business area. (When implemented it will be the Order Entry System; but we will focus on identifying processes at this stage.) Similarly, PRODUCT SUPPLIER is an intersecting entity that represents the *Product Supply Process* used in Purchasing. PRODUCT NEED is the *Product Development Process* used in the Product Development area. Finally, CUSTOMER NEED is the *Customer Needs Analysis Process* used in Marketing.

The data model in Figure 4 is common to many organizations and industries. I will use it to illustrate the principles of reengineering opportunity analysis. For example, by inspection we can already see re-engineering opportunities to integrate some functions based on our understanding of the business. But what of other business areas where mandatory rules have been defined that we are not aware of? How can we ensure that these mandatory rules are correctly applied in our area of interest? The complexity of even this simple data model demands automated entity dependency analysis.

Reengineering analysis of the data model in Figure 4 was carried out by an I-CASE tool that fully automates entity dependency analysis for Business Re-Engineering ([Visible Advantage](#)). The results are shown in Figure 5, an extract from the Cluster Report produced by entity dependency analysis of the data model in Figure 4.



Key to Association Symbols and Meaning

---- -	The data entity at the other end of the association must be associated with <i>one and only one</i> occurrence of the entity at this end.	----0<	The data entity at the other end of the association may be associated with <i>zero, one or many</i> occurrences of the entity at this end.
--------	---	--------	--

---- <	The data entity at the other end of the association <i>must</i> be associated with <i>at least one or many occurrences</i> of the entity at this end.	----0 -	The data entity at the other end of the association <i>will eventually</i> be associated with <i>one and only one occurrence</i> of the entity at this end.
---- 0-	The data entity at the other end of the association <i>may</i> be associated with <i>one and only one occurrence</i> of the entity at this end.	----0 <	The data entity at the other end of the association <i>will eventually</i> be associated with <i>at least one or many occurrences</i> of the entity at this end.

Figure 4: Sales and Distribution data map showing the integration of Order Entry, Purchasing, Product Development and Marketing functions.

Each potential function, process or system represented by an intersecting entity (as discussed above) is called a *Cluster*. Each cluster is numbered and named, and contains all data and processes required for its correct operation. A cluster is thus self-contained: it requires no other mandatory reference to data or processes outside it. Common, inter-dependent, or instead mandatory, data and processes are automatically included within it to ensure its correct operation.

Figure 5 shows Cluster 2, representing the *Order Entry Process*. It has been automatically derived from the data model by *Visible Advantage*. Each of these clusters addresses a business process and is a potential sub-project; common data and processes appear in all clusters that depend on the data or process. The intersecting entity that is the focus of a cluster appears on the last line.

Business Re-Engineering and the Internet Cluster Report
Dec 2 10:00:00 2005 Page 1

-
1. CUSTOMER NEEDS ANALYSIS PROCESS (derived)
 - 1)CUSTOMER
 - 1)NEED
 - 2)CUSTOMER NEED (CUSTOMER NEEDS ANALYSIS PROCESS)

 2. ORDER ENTRY PROCESS (derived)
 - 1)SUPPLIER
 - 2)PRODUCT SUPPLIER (PRODUCT SUPPLY PROCESS)
 - 1)NEED
 - 2)PRODUCT NEED (PRODUCT DEVELOPMENT PROCESS)
 - 1)PRODUCT
 - 2)CUSTOMER NEED (CUSTOMER NEEDS ANALYSIS PROCESS)
 - 1)CUSTOMER
 - 2)ORDER
 - 3)ORDER PRODUCT (ORDER ENTRY PROCESS)
-

Figure 5: Entity dependency analysis of the integrated Sales and Distribution data map from Figure 4, showing the prerequisite processes for the Order Entry Process.

An intersecting entity indicates an process; the name of that process is shown in brackets after the entity name. The intersecting entity on the last line of the cluster (ORDER PRODUCT in Figure 5) is called the "cluster end-point". It is directly dependent on all entities listed above it that are in **bold**: it is inter-dependent on those above it that are not in bold; these represent prerequisite processes of the cluster end-point process. Inter-dependent entities represent common data and processes that are also shared by many other clusters. Thus we can see that the *Order Entry Process* (the cluster end-point) depends on the prerequisite processes: *Product Supply Process*, *Product Development Process* and *Customer Needs Analysis Process*.

Figure 6 next shows that the first two of these processes are fully inter-dependent: a product supplier cannot be selected without knowing the needs addressed by the product (as each supplier names its products differently to other suppliers).

Notice that each entity in Figures 5 and 6 is preceded by a right-bracketed number: this is the project phase number of the relevant entity in the process. Shown in outline form above for each cluster, it represents a conceptual Gantt Chart - the Project Plan for implementation of the process. This Project Plan is automatically derived by *Visible Advantage*, also by using entity dependency analysis.

A cluster in outline form can be used to display a data map automatically. For example, vertically aligning each entity by phase, from left to right, shows the data map in Pert Chart format. Or instead entities can be horizontally displayed by phase, from top to bottom, in an Organization Chart format. An entity name is displayed in an entity box; the attribute names may also be displayed in the entity box. And because the data map is generated automatically, it can be displayed using different data modeling conventions: such as IE notation, or instead IDEF1X.

This ability to automatically generate data maps in different formats is a characteristic of the latest generation CASE tools: data maps can be displayed from clusters. They do not have to be manually drawn; they can be automatically generated. When new entities are added, or associations changed, affected data maps are not changed manually: they are automatically regenerated. Similarly, process maps can be generated from data models. For example, data access processes (Create, Read, Update, Delete), that operate against entities as reusable methods, can be automatically generated as object-oriented process maps by such I-CASE tools.

Business Re-Engineering and the Internet **Cluster Report**
Dec 2 10:00:00 2005 **Page 2**

-
3. **PRODUCT DEVELOPMENT PROCESS (derived)**
 1) **SUPPLIER**
 2) **PRODUCT SUPPLIER (PRODUCT SUPPLY PROCESS)**
 1) **PRODUCT**
 1) **NEED**
 2) **PRODUCT NEED (PRODUCT DEVELOPMENT PROCESS)**
4. **PRODUCT SUPPLY PROCESS (derived)**
 1) **NEED**
 2) **PRODUCT NEED (PRODUCT DEVELOPMENT PROCESS)**
 1) **PRODUCT**
 1) **SUPPLIER**
 2) **PRODUCT SUPPLIER (PRODUCT SUPPLY PROCESS)**
-

Figure 6: Further Entity dependency analysis of Figure 4, showing that the Product Development Process and Product Supply Process are both inter-dependent.

So why have these processes all been included in the cluster in Figure 5 for the *Order Entry Process*? These IE CASE tools also provide direct assistance for Business Re-Engineering.

Identifying Cross-Functional Business Processes from Data Models

We saw in Figure 4 that a **PRODUCT** must have at least one **SUPPLIER**. Figure 5 thus includes the *Product Supply Process* to ensure that we are aware of alternative suppliers for each product. But where did the

Product Development Process and Customer Needs Analysis Process come from?

The data map in Figure 4 shows the business rule that each PRODUCT must address at least one NEED relating to our core business. Similarly the data map follows the Marketing rule that each CUSTOMER must have at least one core business NEED. The *Product Supply Process*, *Product Development Process* and *Customer Needs Analysis Process* have therefore all been included as prerequisite, inter-dependent processes in Figure 5.

The sequence for execution of these processes is shown in Figure 7. This shows each cluster as a named box, for the process represented by that cluster. Each of these process boxes is therefore a sub-project for implementation. This diagram is called a *Project Critical Path Map* as it suggests the development sequence for each sub-project that implements each relevant process.

We can now see some of the power of entity dependency analysis: it automatically applies business rules across the entire enterprise. It is a business expert: aware of all relevant business facts. It determines if other business areas should be aware of relevant business rules, data and processes. It derives a Project Critical Path Map for clear project management of each process sub-project needed to implement those processes as potential computer systems.

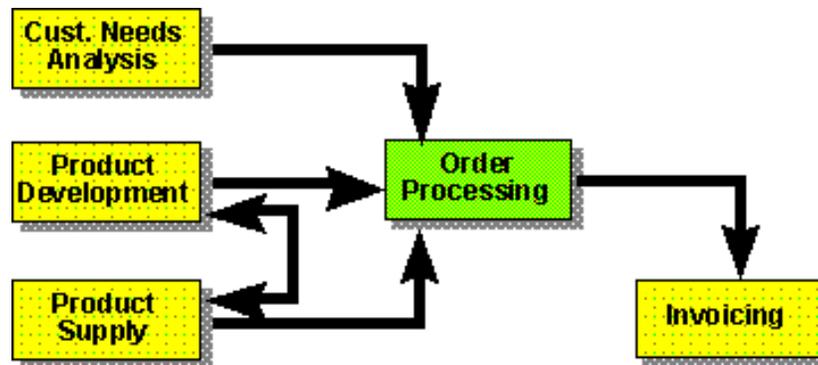


Figure 7: The Order Entry Process depends on prerequisite, inter-dependent processes to its left. These suggest re-engineering opportunities for Order Entry.

So what do these prerequisite, inter-dependent processes suggest? How do they help us to identify re-engineering opportunities? And how can we use the Internet?

Re-Engineering Opportunity Analysis

The Project Critical Path Map in Figure 7 is also used for re-engineering opportunity analysis. It enables us to identify re-engineering opportunities.

Figure 7 shows that the prerequisite processes for Order Entry Process are cross-functional; these separate processes can be integrated. Consider the following scenario for the Order Entry Process - *before Business Re-Engineering*:

Customer: "Customer 165 here. I would like to order 36 units of Product X."

Order Clerk: "Certainly, Sir. ... Oh, I see we are out of Product X at the moment. I'll check with the Warehouse. I will call you back within the hour to let you know when we can expect more of Product X into stock."

Customer: "No don't bother, I need to know now. Please cancel the order."

Clearly, this example shows that the Order Clerk has no access to the Inventory Control System in the Warehouse. There is no way to determine when outstanding purchase orders for out-of-stock products will be delivered. It requires a phone call to the Warehouse staff to get that information. A call-back in an hour is no longer responsive for today's customers. The sale was therefore lost.

Now consider the same scenario - *after Business Re-Engineering*:

Customer: *"Customer 165 here. I would like to order 36 units of Product X."*

Order Clerk: *"Certainly, Sir. ... Oh, I see we are out of Product X at the moment. One moment while I check with our suppliers. ... Yes, we can deliver 36 units of Product X to you on Wednesday."*

What has happened in this scenario? Product X was out of stock, so the *Product Supply Process* then automatically displayed all suppliers of Product X. The Purchasing function had been re-engineered so the Order Clerk can link directly into each supplier's inventory system to check the availability and cost of Product X for each alternative source of supply. For the selected supplier, the Clerk placed a purchase order for immediate shipment and so could confirm the Wednesday delivery date with the customer.

But there are problems with this approach, due to incompatibilities between the supplier's Inventory Control System and the Order Entry System. There may be incompatibilities between the Operating Systems, Data Base Management Systems, LANs, WANs and EDI data formats used by both organizations. We will discuss these problems and their resolution using the Internet, shortly.

The re-engineered *Product Supply Process* discussed above seems revolutionary, but other industries which also take orders online consider this inter-enterprise approach to Order Entry the norm.

Consider the Travel Industry. We phone a travel agent to book a flight to Brisbane (say) because we have business there. We need to fly up on Wednesday evening, for business on Thursday and Friday. But we also decide to take the family and plan to stay for the weekend, returning Sunday evening. The travel agent uses an Airline Reservation terminal to book seats on suitable flights. These are ordered from an inventory of available seats offered by relevant suppliers: the Airlines.

Let us now return to the customer on the phone - still talking to the Order Clerk, who says:

Order Clerk: *"By the way, do you know about Product Y. It allows you to use Product X in half the time. I can send you 36 units of Y as well for only 20% more than your original order. If you agree, we can deliver both to you on Wednesday."*

Customer: *"OK. Thanks for that suggestion, and Wednesday is fine. I look forward to receiving 36 units each of Products X and Y on that day."*

The *Product Development Process* displayed related products that met the same needs as Product X. This suggested that Product Y may be of interest. An order for Y, based on the current order for X, was automatically prepared and priced - and Y was in stock. This extension to the order only needed the customer's approval for its inclusion in the delivery.

What happened here? We see that the *Product Development Process* has also been re-engineered. The ordered Product X satisfies certain needs (see PRODUCT NEED in Figure 4). Other products may also satisfy those needs, as indicated by Product Y above.

Once again, this is commonplace in the Travel Industry. The travel agent knows the customer will be in Brisbane over several nights and so asks whether any hotel accommodation is needed. If so, a booking is made at a suitable hotel using another supplier's system: Hotel Reservations.

The customer continues with the Order Clerk, who now says:

Order Clerk: *"We find that customers using Product X also enjoy Product Z. Have you used this? It has the characteristics of and costs only Can I include 36 units of Product Z as well in our Wednesday delivery?"*

Customer: *"Yes. Thanks again. I confirm that my order is now for 36 units each of Products X, Y and Z - all to be delivered on Wednesday."*

Finally, the *Customer Needs Analysis Process* knew that customers in the same market as Customer 165, who also used Products X and Y, had other needs that were addressed by Product Z. A further extension to include Z in the order was automatically prepared and priced. Z was also in stock and was able to be included in the delivery, if agreed.

This is analogous to the Travel Agent asking if a rental car, and tour or theatre bookings were also needed: likely if a family is in Brisbane, and thus near the Gold Coast tourist resorts for a weekend.

Instead of waiting for stock availability from the Warehouse in the first scenario based on separate, non-integrated processes for each function, the re-engineered scenario let the Clerk place a purchase order directly with a selected supplier so that the customer's order could be satisfied. And the Product Development and Customer Needs Analysis processes then suggested cross-selling opportunities based first on related products, and then on related needs in the customer's market.

Re-engineered, cross-functional processes identified using entity dependency analysis can suggest reorganization opportunities. For example, inter-dependent processes may all be brought together in a new organization unit. Or they may remain in their present organization structure, but integrated automatically by the computer only when needed - as in the re-engineered scenario discussed above.

But what about the incompatibilities we discussed earlier with inter-enterprise access to suppliers' Inventory Systems? The Internet offers us dramatic new ways to address these otherwise insurmountable incompatibilities.

Back to [Contents](#).

THE STATUS OF INTERNET AND INTRANET TECHNOLOGIES

The Internet has emerged since 1994 as a movement that will see all businesses inter-connected in the near future, with electronic commerce as the norm.¹ Let us review the status of Internet and Intranet technologies today, as summarized in Box 1. This indicates that most DBMS and Client/Server tools will interface directly and transparently with the Internet and Intranet. Web browsers, Java, HTML, the Internet and Intranet will all provide an open-architecture interface for most operating system platforms. Previous incompatibilities between operating systems, DBMS products, client/server products, LANs, WANs and EDI disappear - replaced by an open architecture environment based on HTML and Java.

Box 1: A Status Update of Internet and Intranet Technologies

- Web browsers are now available for all platforms and operating systems, based on an open architecture interface using HyperText Markup Language (HTML). A key factor influencing future computing technologies will be this open architecture environment.
- The Web browser market will be largely shared between Microsoft and Netscape. But the strategy adopted by Microsoft will see it rapidly gain market share at the expense of Netscape: it will use its desktop ownership to embed its browser technology (Internet Explorer) as an integral and free component of Windows NT and Windows 98.
- The Internet is based on TCP/IP communications protocol and Domain Naming System (DNS). Microsoft, Novell and other network vendors recognise that TCP/IP and DNS are the network standards for the Internet and Intranets. This open architecture network environment benefits all end-users.
- The battle to become THE Internet language - between Java (from Sun) and ActiveX (from Microsoft) will likely be won by neither. Browsers will support *both* languages, and will automatically download from Web servers, as needed, code in either language (as "applets") for execution. Instead, the winners of this battle will again be the end-users, who will benefit from the open architecture execution environment.
- Data Base Management System (DBMS) vendors (those that plan to survive) will all support dynamic generation of HTML for browsers, with transparent access to the Internet and Intranets by applications using these tools. They will accept HTML input direct from Web forms, process the relevant queries and generate dynamic HTML Web pages to present the requested output. DBMS products with this capability include: Microsoft SQL Server, IBM DB2, Oracle, Sybase, CA-OpenIngres and Informix.
- Client/Server vendors (again those that plan to survive) will also provide dynamic generation of HTML for browsers that will be used as clients, with transparent access to the Internet and Intranets for applications built with those tools. Client code, written in either ActiveX or Java, will both be supported and downloaded as needed for execution, and for generation of dynamic HTML output to display transaction results. Products include: Microsoft Visual Basic 5.0, Visual J++, Access 97; Powersoft Optima++ and Powerbuilder; Centura and SQLWindows; Borland Latte, Delphi & C++.
- Data Warehouse and Data Mining products will provide a similar capability: accepting HTML input and generating HTML output if they are to be used effectively via the Intranet and Internet. And also Screen Scraper tools that provide GUI interfaces for Legacy Systems will become internet-aware: accepting 3270 data streams and dynamically translating them to, or from, HTML to display on the screen. Thus they will provide a transparent HTML interface for easy migration of 3270 mainframe Legacy Systems to the Internet and Intranets.

The open-architecture environment enjoyed by the audio industry - where any CD or tape will run on any player, which can be connected to any amplifier and speakers - has long been the holy grail of the IT industry. Finally, once the industry has made the transition over the next few years to the open-architecture environment brought about by Internet and Intranet technologies, we will be close to achieving that holy grail !!!

The client software will be the web browser, operating as a "fat" client by automatically downloading Java or ActiveX code when needed. Client/server tools will typically offer two options, each able to be executed by any terminal which can run browsers or HTML-aware code:

1. Transaction processing using client input via web forms, with dynamic HTML web pages presenting output results in a standard web browser format, or
2. Transaction processing using client input via client/server screens, with designed application-specific output screens built by client/server development tools. This optional client environment will recognise HTML, dynamically translating and presenting that output using the designed application-specific screens.

These client/server development tools will provide transparent access to data base servers using HTML-access requests, whether accessing operational data or Data Warehouses. In turn the data base servers will process these requests - transparently using conventional languages, Java or ActiveX to access new or legacy data bases as relevant. These may be separate servers, or instead may be mainframes executing

legacy systems.

Web servers will then operate as application servers, executing Java, ActiveX or conventional code as part of the middle-tier of three-tier client/server logic distribution, with data base servers also executing Java, ActiveX or conventional code as the third logic tier.

Development will be easier: many of the incompatibilities we previously had to deal with will be a thing of the past. Open architecture development using the technologies of the Internet will also be part of the Intranet: able to use any PC and any hardware, operating system, DBMS, network, client/server tool or Data Warehouse. This will be the direction that the IT industry will take for the foreseeable future.

Back to [Contents](#).

BUSINESS RE-ENGINEERING AND THE INTERNET

We now see that the rush to implement systems based on Internet and Intranet technologies will resolve the incompatibility problems we discussed earlier. HTML will become the standard interface between the Order Entry system and the Suppliers' systems.

Suppliers are providing a capability for the world to order products via the Internet, using Web forms to make these order requests. These Web form transactions are sent from the customer's browser to the URL (Uniform Resource Locator) of the supplier's web server, with a data string of the field names and contents entered by the customer on the Web form. The input transaction is processed by the supplier's Inventory system and ordered products are shipped to the nominated delivery address.

Thus the Order Entry System implemented for the Order Entry Process discussed earlier can construct a data string as if it was a transaction entered from a Web form. This computer-generated transaction can then be sent to the URL of each supplier of an out-of-stock product, to satisfy the order of the Order Entry Dept's own customer and so generate a Purchase Order with a selected supplier - who now can deliver directly to our customer. Thus the earlier incompatibility problem, in automatically accessing suppliers' systems, disappears.

New reengineering opportunities emerge from immediate access to customers and suppliers via the Internet. But this also means that *the chaos of redundant data that exists in most enterprises ... will now be visible to the world!* If this redundant data problem is not resolved and new re-engineered process implemented, as discussed in this paper, the chaos will now be apparent from the front window of each organization's web site. Not by what can be done, but rather by what they cannot do when compared with competitors. Customers will therefore leave with the click of a mouse, and go to those competitors that can and will offer them the service they demand.

Back to [Contents](#).

CONCLUSION

To re-engineer only by improving processes using Business Process Re-engineering (BPR) is like closing the barn door after the horse has bolted! Existing processes must be related back to business plans. Only those processes that support plans relevant to the future should be considered for re-engineering. If a process is important and there are no plans today to guide the process, then plans must be defined that will provide the needed process guidance for the future. If this is not done, then BPR has the same long term impact on an organization's competitiveness as rearranging the deckchairs had on the survival of the Titanic.

Business plans include policies, goals, objectives, strategies and tactics. They may need information for decision-making that is not presently available in the enterprise. This information may be derived from data that does not exist today. Thus no processes will presently exist to provide the information, or that maintain non-existent data up-to-date. By looking at processes only, BPR may never have seen the need for this new information, data and processes.

However the business plans provide a catalyst for definition of data and information in data models defined at each relevant management level. These data models are analysed automatically by entity dependency to determine inter-dependent processes. In turn, these suggest cross-functional re-engineered processes. Entity dependency analysis derives the project plans automatically that are needed to implement the data bases and systems required by these re-engineered processes.

Only when all three apexes are addressed in the triangle of Figure 3 can Business Re-Engineering fully consider the needs of the business for the future. These are the three steps to success in BRE. Only then can re-engineered organizations be built that are effective, efficient, best-of-breed, and able to compete aggressively in the future.

New reengineering opportunities emerge from immediate access to customers and suppliers via the Internet. But this also means that *the chaos of redundant data that exists in most enterprises ... will now be visible to the world!* If this redundant data problem is not resolved and new re-engineered process implemented, as discussed in this paper, the chaos will now be apparent from the front window of each organization's web site. Not by what can be done, but rather by what they cannot do when compared with competitors. Customers will therefore leave with the click of a mouse, and go to those competitors that can and will offer them the service they demand.

Back to [Contents](#).

REFERENCES AND GLOSSARY

Business-Driven IE:

[Information Engineering](#) (IE) - first developed in Australia and New Zealand in the late 1970s - is a dominant systems development methodology used world-wide. At that time IE was DP-driven. Business-driven IE is a more powerful variant of Information Engineering - developed in the mid 1980s for the rapid-change environment of the 1990s. The business-driven IE variant is the focus of this paper.

Data Attribute:

Data attributes provide information about the data entity in which they reside. Attributes are physically implemented as Columns in a Table.

Data Entity:

A data entity represents data that is stored for later reference. It is a logical representation of data, implemented physically as a Table in a relational data base environment.

Data Model Analysis:

Many CASE tools still support only DP-driven IE (developed in the 70s to support the 80s) and use Affinity Analysis. Unfortunately Affinity Analysis does not provide the analytical rigour that is needed to identify cross-functional processes, which are vital for success with Business Re-Engineering in the 90s. Entity Dependency analysis is a more powerful, objective method used to identify cross-functional processes and derive project plans automatically from data models.

Finkelstein, 1981a:

[Clive Finkelstein](#), "Information Engineering", Series of six InDepth articles published by Computerworld, Framingham: MA (May - June 1981).

Finkelstein et al, 1981b:

[Clive Finkelstein](#) and James Martin, "Information Engineering", Savant Institute, Carnforth, Lancs: UK (Nov 1981).

Finkelstein, 1989:

[Clive Finkelstein](#), "An Introduction to Information Engineering", Addison-Wesley, Reading: MA (Nov 1989) [ISBN 0-201-41654-9].

Finkelstein, 1992:

[Clive Finkelstein](#), "Information Engineering: Strategic Systems Development", Addison-Wesley, Reading: MA

(1992) [ISBN 0-201-50988-1].

Finkelstein, 1996:

[Clive Finkelstein](#), "[The Competitive Armageddon: Survival and Prosperity in the Connected World of the Internet](#)", Information Engineering Services, Melbourne: Australia (Nov 1996). This can be downloaded from the "[White Papers](#)" section of the IES Web Site at - <http://www.ies.aust.com/~ieinfo/>

Hammer, 1990:

Michael Hammer, "[Reengineering Work: Don't Automate, Obliterate](#)", Harvard Business Review, Cambridge: MA (Jul-Aug 1990).

IE CASE:

These software products are [Visible Advantage](#) (an Integrated I-CASE tool that automates business-driven IE) and [Visible Advisor](#) (a hypermedia methodology reference product for business-driven IE). They can be used to design and build systems for any hardware or software platform environment.

Visible Advantage:

[Visible Advantage](#) is an I-CASE tool for Windows 3.1, Windows 95 and Windows NT. It automatically uses entity dependency to analyse data models, identify cross - functional process opportunities for business re-engineering, and derive project plans from data models.

McClure, 1993:

Stephen McClure, "[Information Engineering for Client/Server Architectures](#)", Data Base Newsletter, Boston: MA (Jul-Aug 1993).

Tapscott et al, 1992:

Don Tapscott and Art Caston, "[Paradigm Shift: Interview with Tapscott and Caston](#)", Information Week (Oct 5, 1992).

Tapscott et al, 1993:

Don Tapscott and Art Caston, "[Paradigm Shift: The New Promise of Information Technology](#)", McGraw-Hill, New York: NY (1993).

Zachman, 1992:

[John Zachman](#), "[Framework for Information Systems Architecture](#)", Zachman International, Los Angeles: CA (1992). This can be downloaded from the "[White Papers](#)" section of the IES Web Site at - <http://www.ies.aust.com/~ieinfo/>

Back to [Contents](#).

AUTHOR

Clive Finkelstein is the "Father" of Information Engineering (IE), developed by him from 1976. He is an International Consultant and Instructor, and the Managing Director of Information Engineering Services Pty Ltd (IES) in Australia.

Clive Finkelstein's books, online interviews, courses and details are available at <http://www.ies.aust.com/cbfindex.htm>.

For More Information, Contact:

Clive Finkelstein, Managing Director
Information Engineering Services Pty Ltd
PO Box 246, Hillarys, Perth WA 6923 Australia

Details: <http://www.ies.aust.com/cbfindex.htm>

Web Site: <http://www.ies.aust.com/>

Phone: +61-8-9402-8300

Fax: +61-8-9402-8322

Email: cfink@ies.aust.com

| [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.