CHAPTER 3- IT THEORY: TOOLS, METHODS, AND APPLICATIONS – I

In the preceding chapter we have seen that IT, if used strategically can have a major impact on the way businesses operate – using IT for strategic advantage implies the application of IT to enable and enhance the core competencies of the business. The degree to which the IT function can influence the infrastructure and processes of the organization to run the business better determine the strategic leveraging of IT within the organization. IT also creates competitive advantages (differentiation) by being a proactive contributor to business-critical functions. For example, it can be used to create new marketing opportunities for the organization by transcending geographical boundaries at negligible cost utilizing web technologies. If the business wants to competitive and strategic advantage, in some cases, it is imperative that the organization in question is at least as fast as its competitors in adopting IT, if not becoming an early adopter itself.

Both the competitive and strategic advantage characteristics can rapidly become necessities if the organization lags behind its competition in terms of technology differentiation. If this happens, then these become defensive measures that must be taken for the organization to survive. This indicates that the strategizing of IT should not take place pre or post the strategizing of business. For example, airlines that were not early adopters of new technologies like on-line ticketing systems – and more recently, web check-ins and Web transactions, found themselves losing significant business. However, even for those airlines that did adopt these technologies, the advantage was typically temporary – lasting only for the time it took for competitors to adopt the same (or improved) versions of the technology. Similar examples can be seen in the banking, financial services and insurance vertical (BFSI). Banks that were early implementers of ATMs (automated teller machines) found that their advantage was short-lived.

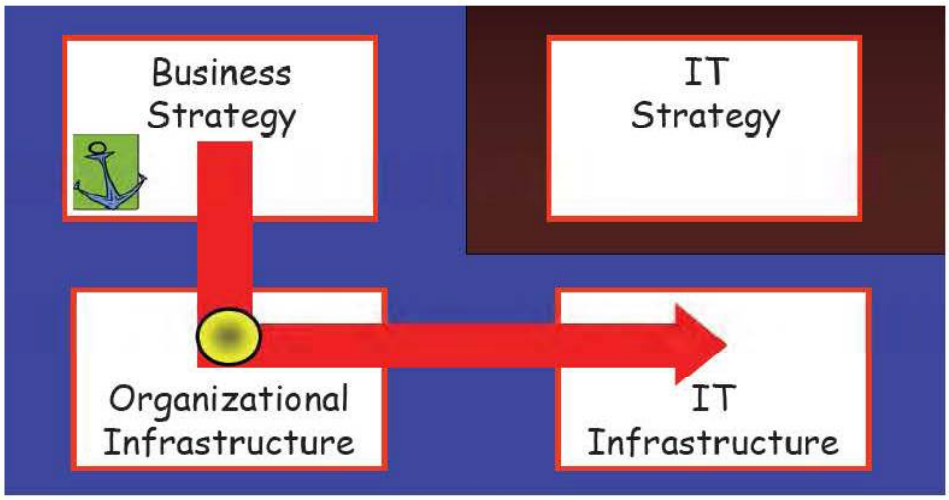
3.1 Methods of Transformation

There are four broad ways of looking at the transformation that has been brought about in these companies – through the business, the technology, the service level, and the strategic view. Each of these views leads to the adoption by an organization of either:

1. IT tools and techniques, or
2. Transformation of business, or
3. Some combination of the two

Certain authors *(Luftman, 2003)* have shown this as a combination of a domain anchor, domain pivot, and impacted domain. The domain anchor is the area that provides or drives the change forces applied to that domain. The domain pivot is the problem area being addressed and the impacted domain is the area being affected by a change to the domain pivot.

Diagram 3.1 - Business strategy is the anchor domain

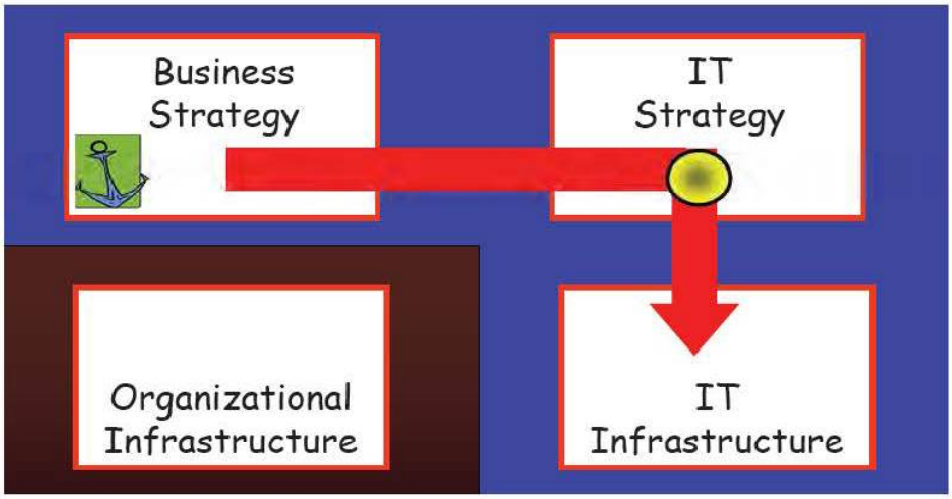


In the above diagram 3.1, business strategy is the anchor domain, the organization's infrastructure is the pivot domain and the impacted domain is the IT infrastructure. This illustrates the strategy execution view. It reflects how emerging technology could influence or

enable new business strategies (e.g., new market opportunities), thus creating competitive advantage.

On the other hand we have the technology potential view, where the focus is on establishing a strategic fit for IT. This perspective is on how IT can be used to enable new business strategies. This is shown in the following diagram 3.2.

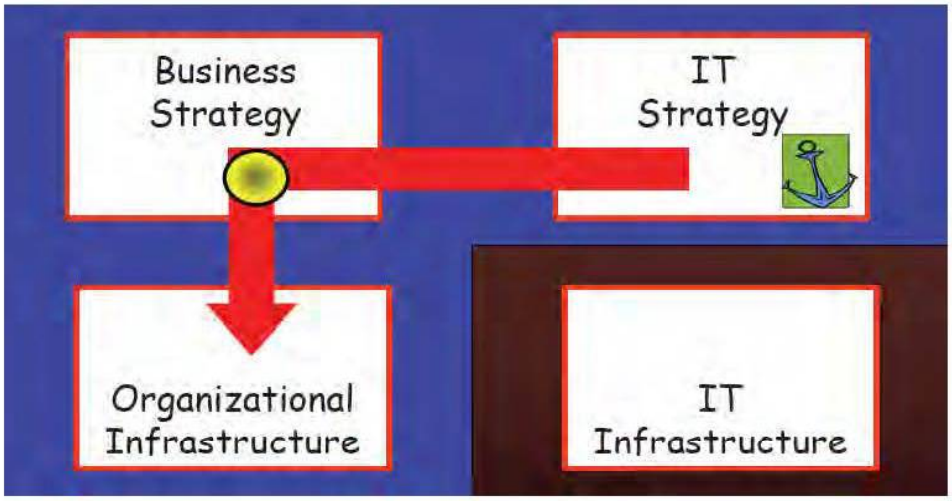
Diagram 3.2 - Technology potential view



Here, while business remains the anchor, strategic IT becomes the pivot that drives the IT infrastructure for the organization.

A third situation exists – which is the competitive potential view shown in the following diagram 3.3.

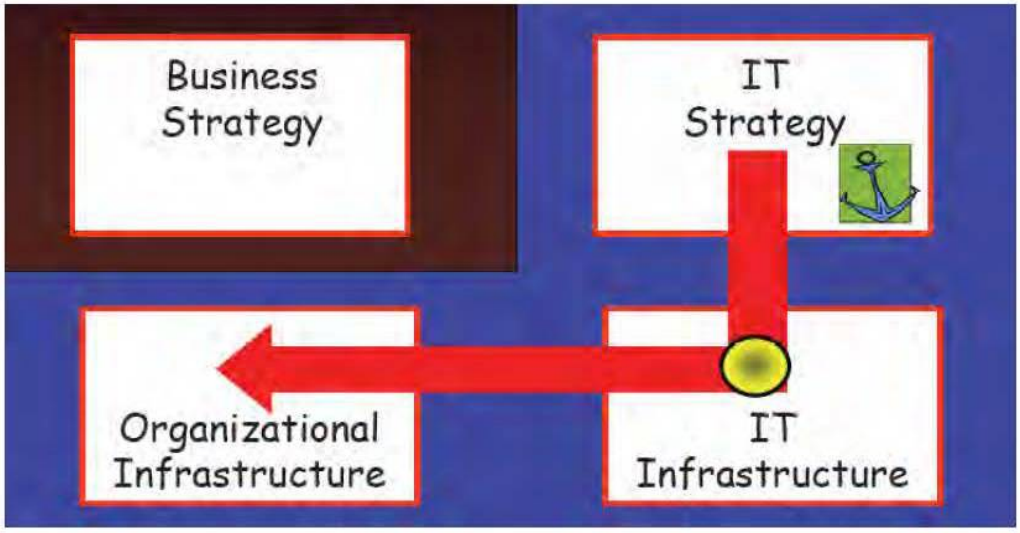
Diagram 3.3 - Competitive potential view



Here, IT strategy becomes the anchor while business becomes the pivot and the impacted domain becomes the organizational infrastructure – which leads to business transformation. IT infrastructure remains stable and unchanged in this context while the others change around it.

Last but not the least, in the service level view; the following happens as shown in diagram 3.4.

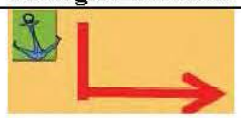
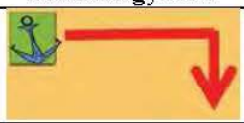
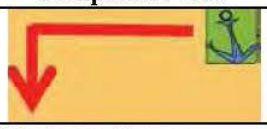
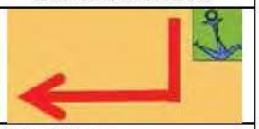
Diagram 3.4 - Service level view



Here IT is applied strategically as the anchor, the IT infrastructure acts as a pivot and the impacted domain is again the organizational infrastructure. This applies in situations where the organization is using a tightly integrated IT environment. What if this environment failed? This view focuses on the ability to deliver information technology products and services to the organization. Since the service business is constantly balancing short-term responses to customer demand (for example, to provide a new feature or report) with the long-term investment in infrastructure (for example, building a network or a security architecture), a major challenge is to establish and sustain a direction or priority set. In this role, the management team uses these priorities to balance short and long-term decisions effectively.

Technology matrix: So far I have been clubbing IT into a generic term that is all-encompassing. However, in reality, this is not the case. IT can be broken into a number of sub-heads – each of which can have more branches – until often we reach a product or manufacturer specialization. For example, if we talk about mail and messaging, at the enterprise level we would typically talk about companies like Sun Microsystems with their Sun ONE messaging server, Microsoft with their Exchange server and IBM with Domino. In addition, we will have a slew of other smaller vendors – including open source products like sendmail on Linux. The features provided by these systems are vastly different. So are their target markets, skill-sets required to manage them, and their scalability and fault tolerance levels.

Table 3.1 - Technology Matrix

	Strategic Execution	**Technology view**	**Competitive view**	**Service oriented**
Nature of domain relationship				
Role of top management	Leader	Technology visionary	Business visionary	Prioritizor
Role of IT management	Functional manager	Technical architect	Business architect	Service manager
IT focus areas	Reactive / Responsive	Enable value-add to business	Drive value-add to business	Business within a business
IT performance criteria	Financial	IT value to product/service	Product/service value from IT	Customer satisfaction
Strategic planning method	BPR of IT planning	IT strategy	Business strategy	IT planning, re-engineering or execution

In this context, I consider some of the IT consulting companies, and discuss the areas in which they operate under the following 17 broad categories. I focus on (1) to (8) in this part and (9) to (17) in the next chapter.

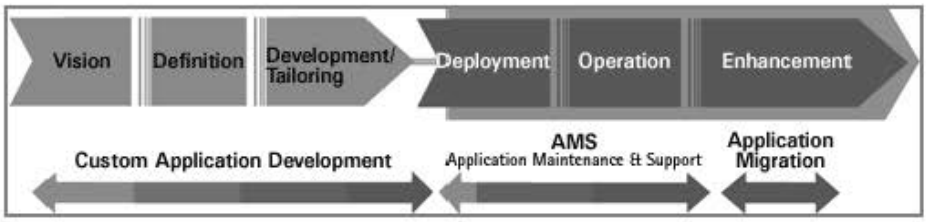
1. Application related services
2. Strategizing services
3. E-enabling services
4. Business Intelligence services
5. Business Process Management (BPM)
6. Portals & Content Management
7. Enterprise Applications Services
8. Security related services
9. Legacy application services including migration
10. Product Life Cycle Management (PLM)
11. Quality services
12. System integration services
13. Enterprise Application Integration
14. Infrastructure related services
15. Testing Services
16. Service Oriented Architecture (SOA)
17. Compliance

Each of these operations areas, in turn, have a number of sub-areas – each of which, again, have their own operations areas – ultimately leading to a specific product, technology, and methodology. While it is not possible to cover exhaustively, all technologies available today, the following attempts to elaborate each of these areas in some measure of detail – at least covering the pre-dominant, mature technology groups.

3.1.1 APPLICATION RELATED SERVICES

These cover the entire lifecycle of application development – from conception to making the application to post-implementation support. This is shown in the following diagram 3.5.

Diagram 3.5 - Lifecycle of Application Development



Normally, each stage is a specialization in itself, with dedicated teams and personnel looking after its functions. A brief explanation of each stage is as follows:

Vision and Definition, often the first step in attempting to design a new piece of software, whether it be an addition to an existing software, a new application, a new subsystem or a whole new system, is, what is generally referred to as vision and definition. Assuming that the developers (including the analysts) are not sufficiently knowledgeable in the subject area of the new software, the first task is to investigate the 'domain', 'industry' or 'vertical' of the software. The more knowledgeable they are about the domain already, the less the work required. Another objective of this work is to make the analysts who will later try to elicit and gather the requirements from the area experts or professionals, speak with them in the domain's own terminology and to understand better what is being said by these people. So, this phase is an important prelude to extracting and gathering the requirements – which typically forms the next stage, which is arguably the most important phase as well. Customers typically know what they want, but not what software should do, while incomplete, ambiguous or contradictory requirements are recognized by skilled and experienced software engineers. Frequently demonstrating live code may help reduce the risk that the requirements are incorrect. Specification is the task of precisely describing the software to be written, possibly in a rigorous way. In practice, most successful specifications are written to understand and fine-tune applications that were already well-developed, although safety-critical software systems are often carefully specified prior to application development. Specifications are most important for external interfaces that must remain stable. As part of the specifications, the software's architecture needs to be defined. The architecture of a software system refers to an abstract representation of that system. Architecture is concerned with making sure the software system will meet the requirements of the product, as well as ensuring that future

requirements can be addressed. The architecture step also addresses interfaces between the software system and other software products, as well as the underlying hardware or the host operating system. Once this is done successfully, the actual coding (programming) can start. The choice of language and platform will correspond to those defined in the system architecture. Reducing a design to code may be the most obvious part of the software engineering job, but it is not necessarily the largest portion.

Upon completion of coding, testing of the software and its constituent parts has to be done to ensure that each component of the software works independently as well as together as a system. This process may be done manually by teams, or by automated testing software that performs the tests within a given set of parameters. An important (and often overlooked) task is documenting the internal design of software for the purpose of future maintenance and enhancement. This is referred to as system documentation, and it is normally written by specialist teams called technical writers. Additionally, user documentation also needs to be done that will tell users how to operate and work with the system. This includes documents like user manuals, performance tuning manuals etc. A parallel and equally important activity is training and support. A large percentage of software projects fail because the developers fail to realize that it doesn't matter how much time and planning a development team puts into creating software if nobody in the organization ends up using it. People are occasionally resistant to change and avoid venturing into an unfamiliar area, so it is very important as a part of the deployment phase, to have training classes for the most enthusiastic software users (build excitement and confidence), shifting the training towards the neutral users intermixed with the avid supporters, and finally incorporate the rest of the organization into adopting the new software. Users will have lots of questions and software problems which lead to the next phase of software.

After all the above stages are completed, the maintenance phase can begin. This encompasses maintaining and enhancing software to cope with newly discovered problems or new requirements. In some cases, this can take far more time than the initial development of the software. Not only may it be necessary to add code that does not fit the original design but just determining how software works at some point after it is completed may require significant effort by a software engineer. About ⅔ of all software engineering work is maintenance, but

this statistic can be misleading. A small part of that is fixing bugs. Most maintenance is extending systems to do new things, which in many ways can be considered new work. In comparison, about ⅔ of all civil engineering, architecture, and construction work is maintenance in a similar way.

3.1.2 STRATEGIZING SERVICES

This subscribes to the view of businesses viewing IT as a strategic asset—a source of both operational excellence and competitive advantage. For these companies, IT is not merely a cost but a critical contributor to the business, focused on improving business value and performance. This is not just limited to cost reduction, but extends to delivering operational efficiencies in a predictable manner as well as being a direct contributor to top and bottom line growth.

The diagram 3.6 below illustrates that successful businesses actually spend less than their lower performing peers on IT because they spend more selectively with emphasis on value creation. The goal shifts from cost containment and reduction to improving the business's operating results. The key is selecting IT investments based on bottom line return. Some of these consulting companies have formed vertical lines of business based on this premise.

Diagram 3.6 - Successful businesses spend less than their lower performing peers on IT

3.1.3 E-ENABLING SERVICES

This is perhaps the most difficult service to qualify as well as quantify – primarily because of the sheer variety and diversity in its application across industries. Everything from web development to designing the infrastructure for an e-commerce initiative to large scale e-governance projects is covered under this head. Thus too, the largest spectrum of technologies and specialty products are in this segment.

This is demonstrated by the figures shown in table 3.2 shown below – in terms of internet penetration especially in Asia which has the highest usage percentage in the world although the penetration on a population base is only 10.2%. Companies and organizations in Asia have been quick to jump on to the internet bandwagon – often leapfrogging complete generations of technologies in the process.

Table 3.2 - World Internet usage and population statistics

World Regions	Population % of World	% Population (Penetration)	Usage % of World	Usage Growth 2000-2005
Africa	14.1 %	2.6 %	2.3 %	423.9 %
Asia	56.4 %	10.4 %	36.5 %	232.8 %
Europe	12.4 %	36.4 %	28.2 %	179.8 %
Middle East	2.9 %	9.6 %	1.7 %	454.2 %
North America	5.1 %	68.6 %	21.8 %	110.4 %
Latin America/Caribbean	8.5 %	14.7 %	7.8 %	350.5 %
Oceania / Australia	0.5 %	52.6 %	1.7 %	134.6 %
WORLD TOTAL	100.0 %	16.0 %	100.0 %	189.0 %

Source: World Internet Usage Statistics

Everything from middleware like IBM's WebSphere, Microsoft's Internet Information Server, or ORACLE's web application server to front end languages like AJAX, active server pages, dot net and j2ee to back end data repositories and database management systems like Microsoft's SQL server, ORACLE's RDBMS, or IBM's DB/2 gets covered under this section. (As the number of participating technologies and product slate is very large, a comprehensive

list has not been given here. Instead, an indicative list of the pre-dominant products in the market space is provided.

It is further not necessary that these services be offered for external audiences only – they could be for purely internal audiences as well – as demonstrated by the number of internal intranet and portal deployments done by organizations worldwide. Some of these deployments have been so successful that they have been subsequently extended to external audiences in extranet implementations. A number of companies like SAP have developed specialized products to cater to this market segment (SAP Enterprise Portal). A number of issues affect these products. Some of these are elaborated below.

- Bandwidth Restrictions and Latency: Slow transmission and methods and large number of accesses to a site in a given time
- User Ignorance and Perceptions: Lack of adequate understanding of the Internet and it usefulness
- Cyberloafing: Surfing the Internet, wasting time and accessing inappropriate materials
- Equity: Inability to access the Web due to the economic and geographical reasons
- Exposure Points: Risk associated with accessing firm's web sites remotely
- Flooding of the Web with content: Including information that is not helpful to the site viewers
- Inadequate search facilities on the WWW: Lack of a high level query language search engines for locating, filtering, and presenting information
- Maintaining and integrity of data: Maintaining up-to-date and accurate information on the site for viewers to use
- Security: Maintaining secure and safe systems and keeping unauthorized user access out
- System incompatibilities: Cross-platform incompatibility that prevent a broad system integrations and access
- Web Performance Tracking: Maintaining account of traffic volumes and utilization of the site and its contents

- Privacy & Confidentiality Agreements: Addressing individual right to privacy and the sharing of confidential information

Each of the above issues has a number of ramifications and will not be discussed here. However, an important consideration – and one that is overriding above all is that these technologies focused around what is termed as Web 1.0 (a term coined from the new generation of web technologies called Web 2.0). Typical characteristics of Web 1.0 sites are that they are for humans to read and understand. There is no obligation of machine processibility. On Web-1.0 pages, the majority of hyperlinks are manually assigned by webmasters who create and maintain these pages. Web-1.0 pages contain only reactive functions or services i.e. they do not contain machine processable semantics, though human readers can well understand their content.

Further, the focus of these pages was static content – with special effects thrown in for good measure to entertain the user. Technologies such as DHTML, Flash and Java applets were used for this purpose. All sites to enable communication with the viewer had to have a 'Contact Us' section on the site – typically containing the address, phone numbers, and possibly email addresses. These sites typically performed functions that were based on conditioned reflexes – in the sense that when the programmed condition was satisfied (for example, a particular field was selected or filled correctly), the responses were pre-determined and predictable. This was subsequently amended to include databases as part of the package, so that a correct selection of elements on a page could fire an SQL query against a relational database and return the results – often in tabular format. The biggest limitation of this technology was that this could not understand remote requests (performed by non-humans), nor could they pro-actively prepare answers for potential questions.

Despite these limitations, the World Wide Web or www was (and continues to be) a great success. Similarly, networks based on the same set of protocols continue to dominate the scenario worldwide in terms of usage.

Web 2.0 basically adds some new technologies – like AJAX (Asynchronous JavaScript and XML) – a web development technique used for creating interactive web applications. The intent is to make web pages feel more responsive by exchanging small amounts of data with

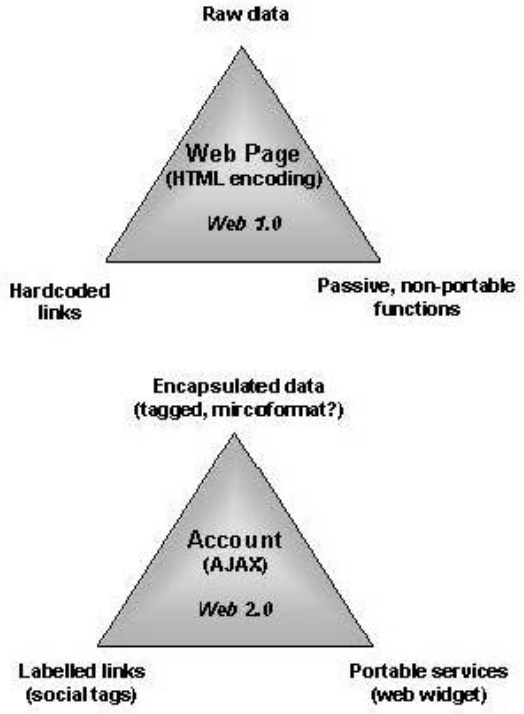
the server behind the scenes so that the entire web page does not have to be reloaded each time the user requests a change. This is intended to increase the web page's interactivity, speed, functionality, and usability.

However, the basis of the success of Web 2.0 lies in two terms – blogging and tagging – the former enhances the character of content and the latter enhances the character of the links one can create on pages. Blogging extends the update of content from personal activities to social activities and tagging enables the creation of hyperlinks from tedious, individual behavior (webmasters) to extremely convenient, mostly automated collaborative behavior. Through the activity of tagging, Web-2.0 based sites teach their own web pages (which could be blogs, or YouTube's personal account web pages containing individual lists of favorites) new knowledge of web facts. Shared tags thus construct implicit hyperlinks among varied web pages. Most of these links could not have been created within the frame of Web 1.0. By tagging a Web-2.0 page (such as a blog), one allows machines to know (or learn) about its content. Blogging behavior then brings the reader directly to the content he/she is interested in.

This is an important consideration in examining why the dot-com bubble burst and consequent downturn in IT actually took place in the new millennium. If one discounts the number of players who did not have an appropriate business model at that time or who weren't serious about their lines of business, there still remains the fact that Web 1.0 content must be generally consumed by humans. Suppose there are 100 million regular web users, and on average every one of them effectively consumes 20 web pages every day. It thus means that a web containing 2 billion pages may satisfy almost all of them. In fact, it allows every of these regular users 5 million days to explore the entire web without repeating a single page. A web bigger than this size becomes generally meaningless to this society because many of them may never been effectively consumed. This was what the bubble 1.0 encountered. The sudden explosion of web content greatly overpowers what regular web users can consume. As a result, many new dot-com sites have so few customers that they cannot survive. But Web 2.0 becomes different. An important phenomenon on Web 2.0 is that machines (such as web blogs) start joining the list of web consumers. Web feeds are typical examples. Due to the help from the machine side, every human user can now effectively consume more web pages every day. For example, web feeds can help every user consume 100 pages every day. It greatly saves

human energy and improves the consuming rate of web content. Therefore, the totally consumable volume of web can be immediately expanded to 12 billion pages. Certainly, real numbers of web users have not been used in this calculation. But this simple calculation as depicted in diagram 3.7 itself shows that Web 2.0 can endure a much larger web volume than Web 1.0.

Diagram 3.7 - Web 2.0 can endure a much larger web volume than Web 1.0



3.1.4 BUSINESS INTELLIGENCE SERVICES:

Prior to the start of the Information Age in the late 20th century, businesses had to collect data from non-automated sources. Businesses then lacked the computing resources to adequately analyze the data, and as a result, executives in companies often made business decisions primarily on the basis of intuition or gut feel. As businesses started automating more and more systems, more and more data became available. However, collection remained a challenge because of a lack of infrastructure for data exchange or to incompatibilities between systems. Analysis of the data that was gathered and reports on the data sometimes took months to

generate. Such reports allowed informed long-term strategic decision-making. However, short-term tactical decision-making continued to rely on intuition. Thus we have business intelligence, a term and a definition that dates to a seminal October 1958 IBM Journal article by Hans Peter Luhn titled A Business Intelligence System.

In modern businesses, increasing standards, automation, and technologies have led to vast amounts of data becoming available. Data warehouse technologies have set up repositories to store these data. Improved Extract, transform, load (ETL) and even recently Enterprise Application Integration (EAI) tools have increased the speed of collecting the data. OLAP reporting technologies have allowed faster generation of new reports which analyze the data. Business Intelligence has now become the art of sifting through large amounts of data, extracting pertinent information, and turning that information into knowledge from which actions can be taken.

Business Intelligence software incorporates the ability to mine data, analyze, and report. Some modern BI software allows users to cross-analyze and perform deep data research rapidly for better analysis of sales or performance on an individual, department, or company level. In modern applications of Business Intelligence software, managers are able to quickly compile reports from data for forecasting, analysis, and business decision-making.

The term Business Intelligence refers to technologies, applications, and practices for the collection, integration, analysis, and presentation of business information and also sometimes to the information itself. The purpose of business intelligence is to support better business decision making. BI describes a set of concepts and methods to improve business decision making by using fact-based support systems. BI is sometimes used interchangeably with briefing books, report and query tools and executive information systems. Business Intelligence systems are data-driven DSS. BI systems provide historical, current, and predictive views of business operations, most often using data that has been gathered into a data warehouse or a data mart and occasionally working from operational data. Software elements support reporting, interactive "slice-and-dice" pivot-table analyses, visualization, and statistical data mining. Applications tackle sales, production, financial, and many other sources of business data for purposes that include, notably, business performance management. For BI to

work effectively, organizations should have secure computer systems which can specify different levels of user access to the data warehouse depending on the level of the user. The system should have sufficient data capacity and a plan for how long data will be stored (data retention). Analysts should set benchmark and performance targets for the system. These are all areas that these consulting companies specialize in and for which skilled manpower is required.

Business Intelligence analysts have developed software tools to gather and analyze large quantities of unstructured data, such as production metrics, sales statistics, attendance reports, and customer attrition figures. Each BI vendor typically develops Business intelligence systems differently, to suit the demands of different sectors (e.g., retail companies, financial services companies, etc.). BI software and applications include a range of tools like AQL - Associative Query Logic, Scorecarding, Business activity monitoring, Business Performance Management and Performance Measurement, Business Planning, Competitive Analysis, User/End-user Query and Reporting, Enterprise Management systems, Executive Information Systems (EIS), Supply Chain Management/Demand Chain Management, and Finance and Budgeting tools. Other BI technologies are used to store and analyze data, such as Data mining (DM), Data Farming, and Data warehouses; Decision Support Systems (DSS) and Forecasting; Document warehouses and Document Management; Knowledge Management; Mapping, Information visualization, and Dashboarding; Management Information Systems (MIS); Geographic Information Systems (GIS); Trend Analysis; Software as a service (SaaS); Business Intelligence offerings (On Demand) — which is similar to traditional BI solutions, but software is hosted for customers by a provider – typically on an Application service provider (ASP) model; Online analytical processing (OLAP) and multidimensional analysis, sometimes called "Analytics" (based on the "hypercube" or "cube"); Real time business intelligence; Statistics and Technical Data Analysis; Web Mining; Text mining; and Systems intelligence. Business intelligence often uses key performance indicators (KPIs) to assess the present state of business and to prescribe a course of action. Examples of KPIs are things such as lead conversion rate (in sales) and inventory turnover (in inventory management).

3.1.5 BUSINESS PROCESS MANAGEMENT

Business Process Management (BPM) is an emerging field of knowledge and research at the intersection between management and IT, encompassing methods, techniques and tools to design, enact, control, and analyze operational business processes involving humans, organizations, applications, documents and other sources of information. The term operational business processes refers to repetitive business processes performed by organizations in the context of their day-to-day operations, as opposed to strategic decision-making processes which are performed by the top-level management of an organization. BPM differs from business process re-engineering, a management approach popular in the 1990s, in that it does not aim at one-off revolutionary changes to business processes, but at their continuous evolution. In addition, BPM usually combines management methods with IT. BPM covers activities performed by organizations to manage and, if necessary, to improve their business processes. While this goal is not new, software tools called business process management systems (BPM systems) have made such activities faster and cheaper. BPM systems monitor the execution of the business processes so that managers can analyze and change processes in response to data, rather than just a hunch. Business Process Management basically, therefore, is a management model that allows the organizations to manage their processes as any other assets and improve and manage them over the period of time. In a medium to large organizations scenario, a good business process management system allows business to accommodate the day to day changes in the business processes due to competitive, regulatory or market challenges in business processes without overly relying IT departments. This strikes a fine balance between dynamic business areas that want to avoid every risk and grab every opportunity on their way through agile changes in their way to business but are very often restricted by a very stable and hard to change IT infrastructure.

3.1.5.1 BPM life cycle

The activities which constitute business process management can be grouped into five categories: Design, Modeling, Execution, Monitoring, and Optimization as depicted in diagram 3.8

Diagram 3.8 - BPM Life Cycle

Process Design: Process Design encompasses the following:

1) (optionally) The capture of existing processes and documenting their design in terms of Process Map / Flow, Actors, Alerts & Notifications, Escalations, Standard Operating Procedures, Service Level Agreements and task hand-over mechanisms

2) Design the "to-be" process covering all the above processes and ensure that a correct and efficient design is theoretically prepared.

A real world analogy can be an "Architect Design" of a house.

Good design reduces the number of problems over the lifetime of the process. Changes to business processes, resulting from changes in the context in which a business operates are a current research area. A Business Process Management Software, ideally, is used to design, model, implement, monitor and optimize human to human, human to system, and system to system workflows which makes evolution of business processes much smoother and close to the regulatory, market, competitive and conformance challenges faced by businesses.

Process Modeling: Process Modeling encompasses taking the process design and introducing different cost, resource, and other constraint scenarios to determine how the process will operate under different circumstances. It also involves running "what-if analysis" on the processes like what if I have 75% of resources to do the same task? or what if I want to optimize my process so I can do the same job in 80% of the original cost? A real world analogy can be "wind-tunnel" test of an airplane or test flights to determine how much fuel it will consume and how many passengers it can carry.

Process Execution: The traditional way to automate processes is to develop or purchase an application that executes the required steps of the process. However, in practice, these applications rarely execute all the steps of the process accurately or completely. Another approach is to use a federation of software and human intervention. Because of the complexity of the federated approach, documenting a process is difficult. This makes changing or improving the process difficult. As a response to these problems, software has been developed that enables the full business process (as developed in the process design activity) to be defined in a computer language which can be directly executed by the computer. The system will either use services in connected applications to perform business operations (e.g. calculating a repayment plan for a loan) or, when a step is too complex to automate, will message a human requesting input. Compared to either of the previous approaches, directly executing a process definition is much more straightforward and therefore easier to improve. However, automating a process definition requires flexible and comprehensive infrastructure which typically rules out implementing these systems in a legacy IT environment. The commercial BPM software market has focused on graphical process model development, rather than text-language based process models, as a means to reducing the complexity of model development. Visual programming using graphical metaphors has increased productivity in a number of areas of computing and is well accepted by users. Business rules have been used by systems to provide definitions for governing behavior, and a business rule engine can be used to drive process execution and resolution.

Process Monitoring: Process monitoring encompasses the tracking of the individual processes so that information on their state can be easily seen and the provision of statistics on the performance of one or more processes. An example of the tracking is being able to determine the state of a customer order (e.g. ordered, arrived, awaiting delivery, invoice paid) so that problems in its operation can be identified and corrected. In addition, this information can be used to work with customers and suppliers to improve their connected processes. Examples of the statistics are the generation of measures on how quickly a customer order is processed, how many orders were processed in the last month etc. These measures tend to fit into three categories: cycle time, defect rate and productivity. The degree of monitoring depends on what information the business wants to evaluate and analyze and how business wants it to be monitored, in real-time or ad-hoc. Here, business activity monitoring (BAM) extend and

expand the monitoring tools in BPMS. Process monitoring includes process mining which is a collection of methods and tools related to process monitoring. The aim of process mining is to analyze event logs extracted through process monitoring and to compare them with an 'a priori' process model. Process mining allows process analysts to detect discrepancies between the actual process execution and the a priori model as well as to analyze bottlenecks.

Process Optimization: Process optimization includes retrieving process performance information from modeling or monitoring phase and identifying the potential or actual bottlenecks and potential rooms for cost savings or other improvements and then applying those enhancements in the design of the process thus continuing the value cycle of business process management.

Although the initial focus of BPM was on the automation of mechanistic business processes, this has since been extended to integrate human-driven processes in which human interaction takes place in series or parallel with the mechanistic processes. A common form is where individual steps in the business process which requires human intuition or judgment to be performed are assigned to the appropriate members of an organization (as with workflow systems). More advanced forms are in supporting the complex interaction between human workers in performing a workgroup task. In this case, many people and system interact in structured, ad-hoc, and sometimes completely dynamic ways to complete one to many transactions. The BPM software supports and monitors these processes as well as permits their ongoing redefinition at runtime. BPMS can be used to understand organizations through expanded views that would not otherwise be available to organize and present. These views include the relationships of processes to each other which, when included in the process model, provide for advanced reporting and analysis that would not otherwise be available. BPM is regarded to be the crucial backbone of enterprise content management.

Not all activities can be effectively modeled and some processes are best left alone. The value in BPMS is not in automating very simple or very complex tasks, it is in modeling processes where there is the most opportunity. And it is precisely here that these consulting companies can show their mettle.

3.1.6 PORTALS AND CONTENT MANAGEMENT

A web portal is a site that functions as a point of access to information on the World Wide Web. Portals present information from diverse sources in a unified way. Aside from the search engine standard, web portals offer other services such as news, stock prices, infotainment and various other features. Portals provide a way for enterprises to provide a consistent look and feel with access control and procedures for multiple applications, which otherwise would have been different entities altogether. A personal portal is a site on the World Wide Web that typically provides personalized capabilities to its visitors, providing a pathway to other content. It is designed to use distributed applications, different numbers and types of middleware and hardware to provide services from a number of different sources. In addition, business portals are designed to share collaboration in workplaces. A further business-driven requirement of portals is that the content be able to work on multiple platforms such as personal computers, personal digital assistants (PDAs), and cell phones.

It is often necessary to have a centralized application that has access to various other applications within the same enterprise to share the information across the applications. Also the various users with different roles accessing the different applications prefer to have a single access point to all of them over the Internet. They like to personalize the applications and have the coupled applications coordinated. Above all, the administrator users like to have administrative tools all in a single place to administer all the applications. All these are achieved through portals. Since all the applications share information through portals, there is better communication between various types of users. Another advantage of portals is that they can make event-driven campaigns. A list of the advantages of using portals is:

- Intelligent integration and access to enterprise content, applications and processes
- Improved communication and collaboration among customers, partners, and employees
- Unified, real-time access to information held in disparate systems

- Personalized user modification and maintenance of the website presentation

In the late 1990s, the web portal was a hot commodity. After the proliferation of web browsers in the mid-1990s, many companies tried to build or acquire a portal, to have a piece of the Internet market. The web portal gained special attention because it was, for many users, the starting point of their web browser. Netscape became a part of America Online, the Walt Disney Company launched Go.com, and Excite and @Home became a part of AT&T during the late 1990s. Lycos was said to be a good target for other media companies such as CBS. Many of the portals started initially as either web directories (notably Yahoo!) and/or search engines (Excite, Lycos, AltaVista, infoseek, and Hotbot among the old ones). Expanding services was a strategy to secure the user-base and lengthen the time a user stayed on the portal. Services which require user registration such as free email, customization features, and chat rooms were considered to enhance repeat use of the portal. Game, chat, email, news, and other services also tend to make users stay longer, thereby increasing the advertising revenue.

The portal craze, with "old media" companies racing to outbid each other for Internet properties, died down with the dot-com flameout in 2000 and 2001. Disney pulled the plug on Go.com, Excite went bankrupt and its remains were sold to iWon.com. Some notable portal sites — Yahoo!, for instance — remain successful to this day. To modern dot-com businesses, the portal craze serves as a cautionary tale about the risks of rushing into a market crowded with highly-capitalized but largely undifferentiated me-too companies. Along with the development and success of international personal portals such as Yahoo!, regional variants have also sprung up. Some regional portals contain local information such as weather forecasts, street maps and local business information. Another notable expansion over the past couple of years is the move into the formerly unthinkable markets. "Local content - global reach" portals have emerged not only from countries like Korea (Naver.com), India (Rediff.com, Indiatimes.com), China (Sina.com), Romania (Neogen.ro), Greece (in.gr) and Italy (Webplace.it), but in countries like Vietnam where they are very important for learning how to apply e-commerce, e-government, etc. Such portals reach out to people across the world.

At the end of the dot-com boom in the 1990s, many governments had already committed to creating portal sites for their citizens. In Delhi for example, the Delhi Government has a portal (delhigovt.nic.in), in addition to portals developed for specific audiences such as incometaxindia.gov.in; in the United Kingdom the main portals are Directgov (for citizens) and businesslink.gov.uk (for businesses). Many Indian states have their own portals – like Uttaranchal ua.nic.in etc. A number of government agencies choose to outsource the running and management of their websites to external agencies. The National Informatics Centre (NIC) and Education Research Network (ERNET) are two such agencies in India that provide these services.

One of the issues that arise with government web portals is that different agencies often have their own portals and sometimes a statewide portal-directory structure is not sophisticated and deep enough to meet the needs of multiple agencies. Over the last few years more sophisticated technologies are allowing loosely-coupled portal integrations bound together by standard interoperability formats (like SIF – an open standard based on XML) that support a more heterogeneous non-parallel environment.

Corporate portals (intranets) gained popularity during the 1990s. Having access to a variety of company information via a web browser was a new way of working. Intranets quickly grew in size and complexity, and webmasters (many of whom lacked the discipline of managing content and users) became overwhelmed in their duties. It wasn't enough to have a consolidated view of company information, users were demanding personalization and customization. Webmasters, if skilled enough, were able to offer some capabilities, but for the most part ended up driving users away from using the intranet. The 1990s were a time of innovation for the concept of corporate web portals. Many companies began to offer tools to help webmasters manage their data, applications and information more easily, and through personalized views. Some portal solutions today are able to integrate legacy applications, other portals objects, and handle thousands of user requests. Today's corporate portals are sprouting new value-added capabilities for businesses. Capabilities such as managing workflows, increasing collaboration between work groups, and allowing content creators to self-publish their information are lifting the burden off already strapped IT departments.

In addition, most portal solutions today, if architected correctly, can allow internal and external access to specific corporate information using secure authentication or Single-Sign-On as well as complementing technologies like SSL-VPN (Secure Socket Later based Virtual Private Network) and Java Specification Request (JSR) 168 standards (that developed in 2001) allowing the interoperability of portlets across different portal platforms. These standards allow portal developers, administrators and consumers to integrate standards-based portals and portlets across a variety of vendor solutions.

Microsoft's SharePoint Portal Server line of products has been gaining popularity among corporations for building their portals, partly due to the tight integration with the rest of the Microsoft Office products. Research by Forrester Research in 2004 shows that Microsoft is the vendor of choice for companies looking for portal server software. In response to Microsoft's strong presence in the portal market, other portal vendors are being acquired, or are challenging their offering. Oracle Corporation, in 2007, released Web Center Suite, a similar product to SharePoint. Web Center Suite has a full line of collaboration tools (blogs, wikis, team spaces, calendaring, email, etc.).

In addition, the popularity of content aggregation is growing and portal solution will continue to evolve significantly over the next few years. The Gartner Group predicts generation 8 portals to expand on the enterprise mash-up concept of delivering a variety of information, tools, applications and access points through a single mechanism. With the increase in user generated content, disparate data silo's, and file formats, information architects and taxonomist will be required to allow users the ability to tag (classify) the data. This will ultimately cause a ripple effect where users will also be generating ad hoc navigation and information flows. As corporate portals gained popularity, a number of companies began offering them as a hosted service. The hosted portal market fundamentally changed the composition of portals. In many ways they served simply as a tool for publishing information instead of the loftier goals of integrating legacy applications or presenting correlated data from distributed databases. The early hosted portal companies such as Hyperoffice.com or the now defunct InternetPortal.com focused on collaboration and scheduling in addition to the distribution of corporate data. As hosted web portals have risen in popularity their feature set has grown to include hosted databases, document management, email, discussion forums and more. Hosted portals

automatically personalize the content generated from their modules to provide a personalized experience to their users. In this regard they have remained true to the original goals of the earlier corporate web portals.

A related process is managing the content on these portals – and this brings us to **content management**. Content management is a set of processes and technologies that support the evolutionary life cycle of digital information. This digital information is often referred to as content or, to be precise, digital content. Digital content may take the form of text, such as documents, multimedia files, such as audio or video files, or any other file type which follows a content lifecycle which requires management.

The digital content life cycle consists of 6 primary phases: create, update, publish, translate, archive and retire. For example, an instance of digital content is created by one or more authors. Over time that content may be edited. One or more individuals may provide some editorial oversight thereby approving the content for publication. Publishing may take many forms. Publishing may be the act of pushing content out to others, or simply granting digital access rights to certain content to a particular person or group of persons. Later that content may be superseded by another form of content and thus retired or removed from use.

Content management is an inherently collaborative process. It often consists of the following basic roles and responsibilities:

- Creator - responsible for creating and editing content.
- Editor - responsible for tuning the content message and the style of delivery, including translation and localization.
- Publisher - responsible for releasing the content for use.
- Administrator - responsible for managing access permissions to folders and files, usually accomplished by assigning access rights to user groups or roles. Admin may also assist and support users in various ways.

- Consumer, viewer or guest- the person who reads or otherwise takes in content after it is published or shared.

A critical aspect of content management is the ability to manage versions of content as it evolves. Authors and editors often need to restore older versions of edited products due to a process failure or an undesirable series of edits. Another equally important aspect of content management involves the creation, maintenance, and application of review standards. Each member of the content creation and review process has a unique role and set of responsibilities in the development and/or publication of the content. Each review team member requires clear and concise review standards which must be maintained on an ongoing basis to ensure the long-term consistency and health of the knowledge base. A content management system is used to manage the content of a Web site. Content management systems are deployed primarily for interactive use by a potentially large number of contributors. For example, commonly used nowadays are wikis, which are a particular type of content management system that typically contains a set of automated processes that support the following features:

- Import and creation of documents and multimedia material
- Identification of all key users and their roles
- The ability to assign roles and responsibilities to different instances of content categories or types.
- Definition of workflow tasks often coupled with messaging so that content managers are alerted to changes in content.
- The ability to track and manage multiple versions of a single instance of content.
- The ability to publish the content to a repository to support access to the content. Increasingly, the repository is an inherent part of the system, and incorporates enterprise search and retrieval.

The content managed includes computer files, image media, audio files, electronic documents and web content. The idea behind a CMS is to make these files available inter-office, as well as over the web. A Content Management System would most often be used as an archive as well. Many companies use a CMS to store files in a non-proprietary form. Companies use a CMS to share files with ease, as most systems use server-based software, even further broadening file availability. Many Content Management Systems include a feature for Web Content, and some have a feature for a "workflow process." "Work flow" is the idea of moving an electronic document along for either approval, or for adding content. Some Content Management Systems will easily facilitate this process with email notification, and automated routing. This is ideally a collaborative creation of documents. A CMS facilitates the organization, control, and publication of a large body of documents and other content, such as images and multimedia resources.

A web content management system is a content management system with additional features to ease the tasks required to publish web content to web sites. Web Content management systems take the following forms:

- a web content management system is software for web site management
- the work of a newspaper editorial staff organization
- a workflow for article publication
- a document management system

3.1.7 ENTERPRISE APPLICATION SERVICES

This range of services normally offer comprehensive processes and support to implement, configure and customize leading enterprise application packages like those from SAP and ORACLE. Such critical applications must be tightly integrated with custom legacy applications and other commercial packages. These consulting companies help organizations replace tangles of disconnected applications with an integrated, best-in-class enterprise system that efficiently supports and interfaces with core business processes. Typically, this category of services falls into the following heads:

- Implementation Services – Helps transform business by redesigning business processes as well as implementing and configuring best-in-class enterprise applications and technology.

- Upgrade Services – Helps keep enterprise software current with product and technology lifecycles, while taking advantage of new features developed for the enterprise system.

- Consolidation Services – Streamlines number of ERP instances running, leading to dramatic business and operational efficiencies.

- Enhancement Services – Helps add new functionality to existing enterprise systems by enabling new modules or by integrating existing modules with other business functions.

The paradigm of Enterprise Application Services lends itself to the doctrine that companies should operate where their core competencies lie. Thus, for an Oil & Gas E&P company, prospecting should be the major concern, not deploying the logistics module of an ERP system. Applications are the lifeblood of any enterprise, supporting every aspect of how business is planned, managed and executed. As enterprises face the need to rapidly respond to customer requirements and competitive issues, these applications come under increasing pressure to support the business, do more for less, and deliver business value. Increasingly complex and integrated application portfolios, high maintenance costs, and scarce applications support skills can significantly distract attention from core business and competitive issues. This presents a great opportunity for these consulting companies to assist their clients – coupled with the best practices in business process management, their services can help organizations accelerate time-to-market, increase ROI and remain competitive in today's evolving marketplaces.

Picking up where enterprise applications leave off, integration projects create whole business processes or logical views that are beyond the scope of any individual system. Nowadays, when it is the key result area of a CIO to explain ROI for technology projects, integration projects have a higher rate of acceptance as they leverage IT assets that are already in place within an

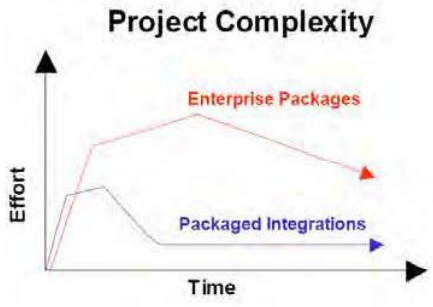
organization. Over time, companies have accumulated myriads of legacy systems – including those resulting from the 1990s-style re-engineering. Thus there is a huge base of standalone or poorly connected systems which one can address. The challenge, of course, is that until recently integration projects were more often than not overtime and over-budget. Their nature makes them extremely labor-intensive, as there are few alternatives to the development and maintenance of custom point-to-point interfaces. In the last five years, software vendors have responded with a variety of new integration solutions, ranging from message brokers to database federation, specific middleware using portals, and business process integration.

One of the fastest growing approaches to integration is for vendors to pre-package business processes that can integrate with existing systems. For clients of these consulting companies, packaged processes promise all the benefits of packaged software, replacing custom integration with technology that is configurable, vendor-supported, upgradeable and stable.

However, the caveat emptor - akin to any other enterprise project is that the benefits result only if implementation is done correctly. Otherwise, the cost of faulty execution includes hastily planned integration processes that must be heavily customized and repeatedly modified after implementation, canceling out the cost advantages of the packaged approach.

Consequently, to ensure success, the first step is to understand the integration process. Through a process of identifying the steps and key stakeholders involved, the path is cleared for developing best practices that in the long run reduce costs and improve rate of return from packaged integration. As can be seen from the following diagram 3.9, compared to enterprise package implementation, package integration projects are far more finite in scope.

Diagram 3.9 - Package integration projects are more finite in scope



With enterprise integration projects, the worlds of package configuration and software engineering converge. Similar to package implementation and configuration, processes have to be defined and software has to be configured and similar to software engineering, the project team requires core analysis, design, development, and testing skills. All this is under a backdrop of implementation across the Enterprise thus increasing the stakes at large. Among integration alternatives, packaged process integration bears the closest resemblance to packaged software because it relies on pre-built, vendor-supported standard functionality. In place of programming, the customer configures vendor developed software through templates and standard workflows, using vendor-supplied interfaces for specific source systems, such as Siebel CRM and Oracle, PeopleSoft and SAP ERP systems. In effect, the customer buys an integration application, or Application Network, rather than a programming tool.

The project team for such projects, in one sense, does bear a close resemblance to that of any enterprise software effort, in that the business and technical sides must be represented. However there are subtle distinctions. On the business side, packaged integrations ideally should be driven by business analysts or process owners who can see "both sides" of a process, as it passes from one system or organization to the next. Naturally, in some organizations, getting someone with the "big picture" of how a particular part of the business works may prove challenging because the views of many process owners often stop at the edge of the organization or the system that they work with, such as SAP. However, in recent years, the emergence of e-commerce initiatives, which tend to span multiple line organizations, has often led to the de facto emergence of process owners who take more global views.

Nonetheless, where departmental boundaries remain solid, designing composite processes is likely to prove a learning experience for business analysts or process owners alike. Frequently, ownership of processes such as order-to-cash or procure-to-pay may be shared among several business analysts or leads from the line organizations. The picture is similar when it comes to the technology aspects of packaged integration projects. Here, people are needed who understand how all applications involved expose data or processes. Maybe the organization is lucky enough to have a person with the global view of how the schemas gel together, but in most cases, this is likely to become a group effort. Out of the project, a single person or group may emerge that takes ownership of data mapping and application communications functions.

There is also an obvious need for a technology lead who learns and owns the third party packaged integrating process solution, and oversees configuration, mapping, installation, testing, and long-term maintenance. Although vendor consultants may initially be brought in to lead the implementation, at some point, at least one application specialist or DBA should eventually take ownership of the packaged integration solution or serve as the liaison for the ongoing support relationship. As enterprises seek to align IT more closely with the business, many are seeking tech leads with business background or vice versa.

Packaged integration projects are usually more narrowly focused than enterprise application implementations because they are more likely to address specific points of pain, rather than transforming the enterprise. Part of this is the nature of the packaged integration solution, which is aimed at specific processes, and part is due to the economic priorities of the times, which favor tactical projects with tangible RoIs over bet-the-enterprise reengineering. Although often undertaken as incremental solutions, packaged integrations cannot be pigeonholed as point solutions because they usually address multiple point processes that traverse organizational boundaries. Additionally, even if the project is conceived as tactical, the impacts can be strategic. For instance, integrating the order-to-cash process may dictate changes over which line organization, and for that matter, which packaged application, "owns" the customer at different stages of the order fulfillment cycle. There are some basic tasks involved in implementing packaged processes. These activities are characterized as tasks rather than steps because, although there may be a logical order (for instance, identifying business need must occur at the start of a project), in some cases multiple tasks (such as risk management, testing, and change management) may be conducted in parallel rather than in sequence.

Task 1 – Identifying the Business Need - Given that packaged integrations address existing systems, the logical conclusion is that the need for projects is established in bottom-up fashion by line organizations that quickly understand the nature of the problem and its solution. However, reality is more complex. Although in some instances line organizations may identify a bottleneck to be automated, in other cases the drivers may be more strategic. At the top of the list is rationalization of systems following merger and acquisition, or the need to simplify a compound process such as order-to-cash. In some cases, the integration need may be obvious:

two organizations merging need to reconcile their product life cycle management systems with Oracle. Other instances, such as streamlining order-to-cash, may only become apparent after an organization studies the major hurdles to goals such as customer retention or margin improvement. But who are the people who get their feet wet at this stage? The answer again is, it depends on the culture of the organization and the level of the initiative. Are the managers hands-on, did the impetus stem from a CEO directive, or do initiatives routinely percolate from the ranks? Another variable is whether business systems are identified up front as the key enablers; if there is awareness that the SAP order entry function could be enhanced, application specialists are also likely to be involved in the discussion this early.

Is this integration therefore an IT or business problem? The conventional wisdom is that the business side drives this part of the process. Surprisingly, when it comes to solidifying the link between a pair of well-identified applications, the impetus may come from IT. The people who need the integration interface are often IT, since they are the ones who must maintain the system. IT staff often press for the integration because the alternative -- having employees rekey the same data twice into two applications -- becomes a source of error and software maintenance problems.

Once an organization completes the integration, gaps, redundancies, or inefficiencies in business processes become more apparent. And, at that point, the business side often seizes the initiative. In some cases, a formal project team may have already been appointed; in others, key movers and shakers from line organizations or top management initiate the process informally, with team formation to follow in the next phase.

Task 2: Requirements Management - Based on the business needs that are established, the project team (if one has already been formally chartered) decides what aspects of the problem are to be addressed. Requirements may be identified in top down fashion (where business users define the services they need) or bottom-up (where software development staff proposes to connect services that are available). In projects that involved packaged integrations, the approach is typically a hybrid because the requirement is to leverage functionality already in place. Consequently, if they weren't involved during business needs identification, application specialists ramp up their involvement now. If a formal project team wasn't formed earlier, it

should be formed at this stage. At this point, project management, business, and technical leads are named, and the cast of supporting players (such as specialists or end users) starts to be identified.

Task 3: Choosing the Solution - The project team, comprising key business analysts who own the processes (or represent the owners), works alongside IT representatives to select the integration approach, and then the vendor. As a fairly standard process for any technology project, the level of IT participation will depend on whether the organization actively enforces overall enterprise architecture, or whether the choice of technology is left to business units or line organizations. Given that packaged process solutions leverage existing applications, in all likelihood application teams are likely to drive selection. And this is where Oracle tools, SAP ABAP, or DBA expertise is required. Any integration solution that involves those core systems must also involve people who know the relevant APIs and schemas. Those are the people who can judge whether the integration solution is technically compatible. If the organization does not have extensive IT skills, it may retain consultants to lead the process. Beyond application skills, the team requires an integration tech specialist as well, because the skills and knowledge sets are different. Packaged processes are the solution of choice where the target applications are known, and where the processes are sufficiently well defined and standardized to provide an adequate market for the integration vendor. Like any packaged application, customers benefit, not from customizing the system, but from the opportunity of avoiding the reinventing of the wheel.

Task 4: Design & Configuration - Now is the time to draw the roadmap, translating requirements into solution. This includes identifying the end state (the integrated business process) and evaluating the underlying technical architecture. At this stage, the team should also specify how it will handle changes, establish goals and metrics for success, and contend with security issues if the project expands access to core systems. The first step in this task is designing integrated business processes. This includes several sub-tasks: identifying the end state, analyzing existing point processes to be linked (and, occasionally, in cases that cannot be avoided, modified), and then specifying the detailed design for the desired business process.

Obviously, it is preferable to design a solution that can be reused. Consequently, while the goal may address a particular pain point or competitive issue, the solution shouldn't be so narrowly defined that it either

1. must be rewritten every time there is any change in the core business, or

2. cannot be applied elsewhere in the organization.

Yet, the solution shouldn't be so generic as to be practically useless. Ideally, rather than modify the integration itself, changes should be restricted as closely as possible to the "edges" of the process within the target application, such as the PLM or CRM systems. It requires somebody with the ability to see the forest through the trees. Naturally, the reality check on designing these super-processes is ongoing feedback from stakeholders (e.g., process owners and end users) -- they cannot be designed in a clean room.

Architecture - Like any software project, implementing packaged processes has a "physical" side. Here, the tech lead works with server, network, and/or database administrators to determine how (and where) the application will be deployed, analyze capacity requirements and impact, and decide whether it will be necessary to add any new infrastructure to handle additional traffic or new utilization patterns generated by the integration processes.

Change Management - Because business doesn't stand still, changes in functionality, goal, or scope are inevitable. Packaged integration projects are no different. Obviously, the goal is to limit change whenever possible, but it is rare in any enterprise software project where initial plans survive unchanged. Consequently, it is preferable to draw a line in the sand up front on how change is to be decided, rather than applied as an ad hoc process after the fact. Ideally, the process should be driven around documents that explain the nature of the change, the justification for it, and its impacts; those documents should be signed off by key stakeholders. Change management requires that downstream impacts are analyzed to the project, processes, and stakeholders; consistent decision making criteria are applied; and that all affected groups are notified when the goals, scope, or other aspects of integration projects depart from plan. It is that simple -- and complex.

Accountability - With ROI more important than ever for IT projects, part of the design phase should incorporate metrics or feedback loops that make the project (and the team responsible for it) accountable. There are many ways to measure success or failure, such as adherence to budgets and deadlines, productivity savings, or the benefits from the ability to deliver new business products or services to the market. On the technology side, success may be measured in quality of service, based on metrics such as performance, availability, and reliability, and the ability to isolate problems away from adjacent systems. For instance, how well is Oracle isolated from problems in the PLM system when the bills of material are integrated? Increasingly, many organizations are demanding "Service Level Agreements" (SLAs) that combine various business and technology metrics in order to get financial approval.

Security – In a post 9/11 world, security is an obvious issue. This is especially true for organizations that are integrating their supply chains. Because integration projects may link, not only applications, but user bases as well, attention must be paid to issues such as access. For instance, in implementing an order-to-cash integration, what are the ramifications when field representatives are given access to warehouse or accounts receivable data? Are they automatically locked out of accounts payable, or would that require changes to SAP user authorizations? The challenge grows even more complex when linking users outside the enterprise. Ideally, security planning should occur alongside the design and configuration process, with extensive input from security specialists or network administrators in the data center.

Task 5: Risk Management - Preferably specified as part of the design phase, the project team must plan how it will brace itself for threats that could compromise existing systems. In addition to the contractual aspects of risk management is the task of preventing existing systems from being corrupted. Consequently, at this stage, the project team must think about what happens when things go wrong, and how impacts will be mitigated. On the technology side of integration projects, the typical answer to risk management is rigorous testing. Ideally, this should be planned during project design, rather than during construction or final operation when unanticipated faults occur.

Task 6: Construction, Testing and Optimization – This is where as they say – the rubber meets the road. In any software project, software is generated based on the original design, tweaked, and redesigned when original blueprints fall short. This stage includes pilots, coding or configuration, plus a cycle of testing of code and data mappings, midcourse corrections, and of course, a full spate of retesting. While at first glance, this process sounds rather complex, with packaged integration approaches, most of the raw coding is eliminated – leaving project teams to focus primarily on configuration and data mapping. There are many approaches to construction, ranging from iterative to "extreme programming" and waterfall development. Regardless of what approach is used, construction should kick off with the well known "conference room pilot," where team members sit down with representative users to test the design on paper. That is followed with a more formal pilot that includes a miniature version of the planned system.

During construction, the group is typically a tag team, involving the integration software specialist, business analyst (and/or user representatives), and in some cases, specialists who maintain the existing enterprise applications. In some projects, the integration specialist may play a double role as expert on the integration software and the packages being linked. Theoretically, packaged integrations should require minimal coding because they pre-build the core transformation and communications logic. However, some additional programming in Java, .NET, or the native scripting of ERP packages, might be necessary to optimize performance out of the linked systems. Ideally, the integrations rely on industry standards, such as those published by OAGi (open applications group inc.), or published native interfaces, such as SAP Business APIs (BAPIs) and BaDIs (business add-ins). However, in some cases, integration teams might take short cuts with unpublished interfaces or hooks to optimize performance. That approach carries significant risks, as package vendors are less obligated to continue supporting private interfaces.

In configuring packaged integrations, much of the focus is on the data schema that is exposed by source and target applications. In most cases, there will be the need for extensive trial and error, because rarely are there exact one-to-one matching of data elements between applications A and B. For instance, the engineering bill of material exposed by a PLM system will likely contain a subset of elements required by ERP, while carrying certain elements

omitted in the financial bill of materials. In some cases, the process of mapping data may bring to light hidden problems in the existing data structures. According to one Sierra customer, the integration project revealed that manufacturing and engineering were keeping similar data under different names and data structures. Consequently, the integration project required significant data cleansing to ensure that both applications (not to mention, line organizations) spoke the same language. In packaged integrations, this is the major form of functional testing.

In some cases, the data mapping process not only reveals inconsistencies, but it also may prompt users to make subtle changes to how they use the existing applications. Even if the connection works once, with other data, errors could recur. Consequently, it is necessary to test and retest multiple permutations of data, a process known as regression testing. Validating multiple permutations of data should be especially familiar to veterans of Electronic Data Interchange (EDI) projects, who are used to fitting square pegs into round holes. The goal is ensuring, not only that the process executes, but that the target system database is not corrupted. Additionally, for integrations involving high volume processes, load testing of back end infrastructure is another key touch point. One of the final pieces, integration testing, occurs just prior to full production, ensuring that parts of the program that worked well on their own don't conflict with each other once brought together. There are technical aspects, such as isolating potential conflicts in database function, and functional conflicts, where the pieces of the process are not assembled together properly. Consequently, this stage involves technical personnel and some end users.

Ideally, all tests should be conducted on separate test beds away from production systems, with cutover only after final integration and functional tests are complete. The test bed would include the integration application plus a representative slice of the source and target databases. In most cases, it will not be practical to replicate the entire production database into the test environment. Depending on the way the project team and IT organization is structured; testing may be performed by developers or application specialists, or by test specialists. IT professionals (which may include members of the project team or test specialists) are responsible for performance and load testing. When validating schema, users may also be involved in developing test cases, and typically, are involved in final functional validations conducted just prior to release to production. The packaged integration will catch some errors,

especially if the data is incomplete or incorrectly formatted. However, other errors will be flagged by obscure error codes from Oracle or other target applications. Most of the time, the corrections must be made to the source systems.

Task 7: Deployment - This is the point where the system goes live. Ideally, the system under development resides on a separate server. In some cases, it could stay on the same machine (which becomes re-designated as production server) with the database pointers redirected to the production system. If there are relatively few integration processes implemented, or if volume is likely to be low or moderate, the entire system could be cut over at once. Otherwise, it is advisable to conduct the rollouts in waves based on whatever is the most logical way of dividing the load (e.g., by region, organization, or application). There is a mix of opinion as to whether final acceptance tests on the production system are necessary. If users are adequately involved during the design, test, and optimization cycle, final acceptance testing might be optional. Training may be part of the deployment process if the project team hands over the system to a separate IT operational group and/or if the integration adds new screens or changes to business processes that impact end users.

Task 8: Compliance - Because technology projects should be accountable, the project life cycle should include a validation cycle after implementation to ensure that the project is meeting the requirements that were enumerated at the start of the effort. Ideally, this should have technical and business elements, both of which could be presented as part of a Service Level Agreement (SLA) that is, in effect, the project team's contract with the enterprise. The contents of SLAs will obviously vary based on the needs of the organization, but it could include business metrics such as process efficiency, productivity, process cycle time. Additionally, it should include technical criteria such as system performance, availability, incidence of bugs or corruption of databases, and impact on overall infrastructure performance and utilization. Ideally, a couple representatives of the original project team from the business and technical sides should be responsible for performing periodic checks to ensure that the system is satisfying original promises. Additionally, the tag team could also document qualitative improvements, such as lead time for introducing key business improvements, or a quantifying of products or services that became possible with the new integration.

Task 9: Post-Implementation maintenance – As any veteran of any enterprise software project will attest, change doesn't stop when the system goes live. But packaged integrations can deliver a powerful advantage by couching the impacts of change. Owing to their powerful transformation engines and configurability, packaged processes can stabilize the touch points between enterprise systems. As competitive requirements or organizational structures change, integration processes will inevitably be revisited. Thanks to their configurability, packaged processes can facilitate change by minimizing impact to source systems. Nonetheless, in the wrong hands, the flexibility of packaged integrations can be problematical if integration teams do not adequately analyze the ramifications of process change. Consequently, the same change management disciplines employed during design and construction should continue to be practiced after the system goes live. Ideally, the "owner" of change management is the business analyst who also owns the integration.

At first glance, the project life cycle closely resembles that for packaged applications. However, there are several critical distinctions. First, packaged integration projects today are typically finite in scope, compared to the enterprise application implementations that preceded them. In place of transforming the enterprise, customers typically choose one or a handful of processes to address tangible pain points. Part of this is attributable to the nature of packaged integration, which offers pre-built solutions addressing specific problems. The other, of course, is the tenor of the times, with enterprises favoring incremental projects with real ROI over the all encompassing re-engineering initiatives of the 1990s.

Another distinction is in the composition of teams. At first glance it appears that packaged integration projects require the same kind of multi-function team of business and technology specialists that delivered enterprise applications. But there is an important difference. The business lead must have a more ecumenical view of the business process, understanding how it works across application boundaries. Similarly, the tech lead should be an expert in the integration and transformation solution, plus the applications that are linked, focusing heavily on schema mapping and identifying the appropriate APIs for the task. Significantly, in a growing number of organizations, IT principals are moving from the business side and vice versa, producing the kind of cross-fertilization that can help packaged integration projects better reaches their potential.

Like any enterprise application project, the proof is in the execution. Given that integration has long had a track record for blown budgets and deadlines, understanding the process is critical in order to build a body of best practices that will change history. Packaged processes are powerful tools that can simplify and reduce the cost of enterprise application integration, if chosen, planned, and deployed properly. With its background developing integrations on an OEM basis for enterprise software vendors such as Oracle, Sierra Atlantic has accumulated dozens of staff years understanding the interfaces and behaviors of major ERP, CRM, PLM, and supply chain systems. It has packaged that experience, with pre-built integrations covering the integration of supply chain management, engineering change orders, bills of materials, and customer-facing processes with leading ERP systems.

3.1.8 SECURITY RELATED SERVICES

This is an area that gained a lot of importance after the 9/11 terrorist attack. Computer security is a branch of information security applied to both theoretical and actual computer systems. Computer security is a branch of computer science that addresses enforcement of 'secure' behavior on the operation of computers. The definition of 'secure' varies by application, and is typically defined implicitly or explicitly by a security policy that addresses confidentiality, integrity and availability of electronic information that is processed by or stored on computer systems.

The traditional approach is to create a trusted security kernel that exploits special-purpose hardware mechanisms in the microprocessor to constrain the operating system and the application programs to conform to the security policy. These systems can isolate processes and data to specific domains and restrict access and privileges of users. This approach avoids trusting most of the operating system and applications.

One use of the term computer security refers to technology to implement a secure operating system. Much of this technology is based on science developed in the 1980s and used to produce what may be some of the most impenetrable operating systems ever. Though still valid, the technology is almost inactive today, perhaps because it is complex or not widely understood. Such ultra-strong secure operating systems are based on operating system kernel

technology that can guarantee that certain security policies are absolutely enforced in an operating environment. This strategy is based on a coupling of special microprocessor hardware features, often involving the memory management unit, to a special correctly implemented operating system kernel. This forms the foundation for a secure operating system which, if certain critical parts are designed and implemented correctly, can ensure the absolute impossibility of penetration by hostile elements. This capability is enabled because the configuration not only imposes a security policy, but in theory completely protects itself from corruption. Ordinary operating systems, on the other hand, lack the features that assure this maximal level of security. The design methodology to produce such secure systems is precise, deterministic and logical.

Systems designed with such methodology represent the state of the art of computer security and the capability to produce them is not widely known. In sharp contrast to most kinds of software, they meet specifications with verifiable certainty comparable to specifications for size, weight and power. Secure operating systems designed this way are used primarily to protect national security information and military secrets. These are very powerful security tools and very few secure operating systems have been certified at the highest level (Orange Book A-1) to operate over the range of "Top Secret" to "unclassified". The assurance of security depends not only on the soundness of the design strategy, but also on the assurance of correctness of the implementation. The Common Criteria quantifies security strength of products in terms of two components, security capability (as Protection Profile) and assurance levels (as EAL levels.) None of these ultra-high assurance secure general purpose operating systems have been produced for decades or certified under the Common Criteria.

In addition, the consulting companies specialize in compliance standards for security like the ISO27002, NIST (National Institute of Standards and Technology). Cyber security standards are security standards which enable organizations to practice safe security techniques in order to minimize the number of successful cyber security attacks. These guides provide general outlines as well as specific techniques for implementing cyber security. For certain specific standards, cyber security certification by an accredited body can be obtained. There are many advantages to obtaining certification including the ability to get cyber security insurance. Further, such certification may be a mandatory condition (as is demonstrated by the

BPO/KPO industry in India) where the client may require such certification prior to awarding the company work.

Additionally, information leak prevention is also high on the list of these companies. Today's security professionals face a daunting challenge of protecting the organization's most valuable asset, its information, amidst widespread investment in new, more efficient communication technologies. As organizations invest in new business systems and processes to exchange critical information to, from and about customers, partners, and employees in real time, more opportunity exists for information leaks. Data breaches are rapidly becoming the forerunner of IT security concerns, in part because of the increase in both the frequency and severity of such breaches. For security professionals, the pressure to provide data security is influenced by three factors:

1. regulatory compliance,
2. protecting confidential data, and
3. mitigating the risk and associated cost of a breach.

Government and industry regulations are arguably the biggest influencers to organizational directives to provide data security. Federal regulations include Sarbanes-Oxley for publicly traded organizations, and the Health Insurance Portability and Accountability Act (HIPAA) for health care organizations, mandate the security of private or confidential information. Information leaks are not solely relegated to organizations with customer data or regulatory requirements; many non-regulated companies share a need to secure sensitive data. Intellectual Property (IP), M&A plans, and other critical assets are strategic to many organizations' success and competitive advantage. These organizations are as concerned about leaks (both external and internal) as regulated companies because of the strategic nature of the information they manage and the frequency with which they fall victim to leaks.

Over the years, organizations have spent a tremendous amount of resources in hopes of protecting their information. However, their efforts have been focused on preventing outsiders from hacking into the organization, educating employees, and securing data at rest. According to analyst firms, the majority of all leaks are the result of unintentional information loss from employees and partners, both external and internal leaks. The average information leak costs organizations approximately $182 per record (according to the Ponemon Institute),

averaging roughly $4,800,000 per breach in total. The high cost of a breach can have a profound effect on organizations P&L, market presence, and competitive advantage as a result of damage to brand and reputation, and loss of customers and IP. As organizations invest millions in business systems increasing the availability of information to build or maintain a competitive edge, there remain a slew of security-related considerations, including:

- Where is the organization's confidential and sensitive data?
- How, where, and when is the data transmitted and by whom?
- How can the data be controlled and protected?
- What is the organization's financial risk (from a leak)?

One of the ways to do this is to use network leakage detection and prevention systems. These are of two types - Network ILD&P and host ILD&P systems. Of these, the network ILD&P, also referred to as gateway-based systems are usually dedicated hardware/software platforms, typically installed on the organization's internet network connection, that analyze network traffic to search for unauthorized information transmissions. They have the advantage that they are simple to install, and provide a relatively low cost of ownership. Because decoding network traffic at high speed is extremely complex and difficult (transmitted objects are broken into small parts, often encoded, and then mixed with other traffic), Network based systems typically integrate with or include technologies to discover information 'at rest' while it is stored in file systems and databases. Discovering sensitive data at rest is far simpler and less time critical, thereby allowing greater levels of accuracy. Taking 'signatures' of data identified at rest, and then looking for such signatures as data passes over the network boundary, is a technique favored by virtually all network system vendors to improve accuracy, and to identify sensitive data that would otherwise be missed.

Host Based ILD&P systems on the other hand, run on end-user workstations or servers in the organization. Like network-based systems, host-based can address internal as well as external communications, and can therefore be used to control information flow between groups or types of users (e.g. 'Chinese walls'). They can also control email and Instant Messaging communications before they are stored in the corporate archive, such that a blocked communication (i.e. one which was never sent, and therefore not subject to retention rules) will not be identified in a subsequent legal discovery situation. Host systems have the

advantage that they can monitor and control access to physical devices (such as USB keys, iPods etc) and in some cases can access information before it has been encrypted. Some host based systems can also provide application controls to block attempted transmissions of confidential information, and provide immediate feedback to the user. They have the disadvantage that they need to be installed on every workstation in the network, cannot be used on mobile devices (e.g. Blackberry), or where they cannot be practically installed (for example on a workstation in an internet café using Outlook Web Access to send corporate e-mail).

Combined systems - The most comprehensive protection is provided by solutions where both Network and Host based systems are combined, preferably within a single management platform where rules are defined centrally and distributed automatically, and where results are gathered and presented in a single console. This approach is endorsed by industry analysts such as Gartner.

In addition, there are a host of security products in the market dealing with all variants of security – be it perimeter protection through firewalls, IDS/IPS systems, UTM (unified threat management) appliances to server file system and application layer protection to desktop client side endpoint security solutions. (In fact, certain companies have even teamed up to provide integrated virus protection at the network level by integrating antivirus software with the operating system on the switches that make up the network.) These products and technologies have their own specialist needs and these consulting companies provide services to fulfill these needs.

To sum up. I have considered in this chapter the methods of transformation, and examined the consulting companies' operations areas, namely, (1) application related services, (2) strategizing services, (3) E-enabling services, (4) business intelligence services, (5) business process management (BPM), (6) portals and content management, (7) enterprise application services, and (8) security related services. I examine nine more operations areas in the next chapter.