THE CHANGING ARCHITECTURE OF GLOBAL WORK: OPPORTUNITIES AND CHALLENGES

A White Paper Prepared for the Keane Workshop

Version: December, 2006

Bala Iyer
N. Venkatraman

Introduction

Consider Tom Friedman's observations:

> "Globalization 3.0 is shrinking the world from a size small to a size tiny and flattening the playing field at the same time. And while the dynamic force in Globalization 1.0 was countries globalizing and the dynamics of Globalization 2.0 was companies globalizing, the dynamic force in Globalization 3.0 – the force that gives it its unique character – is the newfound power for *individuals* to collaborate and compete globally."

This global sharing of expertise and work is made possible through a flat-world platform [1]. As a result, industries such as IT services are becoming global with vendors operating offices all over the world to tap into the expertise that is distributed globally. It has also allowed companies to locate their back-offices in different parts of the world to take advantage of a 24-hour working day. Many Fortune 500 firms with offices and production facilities all over the world are beginning to expect their vendors to support them globally. This has forced vendors to create global delivery models to keep pace with their customer's expectations.

The shift of services work towards a global market model is not unexpected. The architecture of work is dependent on what is tradable. Manufacturing dominated trade until recently. Now it is the era of services. Technology, especially communications and transportation, determines what services can be traded internationally. As a result of investments in global electronic networks, more IT services are tradable than were tradable in the past. In short the implications of the flat-world are that companies are tapping into the global workforce linked together by powerful information technology infrastructure. And the IT service providers need to organize globally to deliver the requisite services at acceptable levels of cost and performance.

The software industry structure has evolved from highly integrated, proprietary solutions to modular, loosely coupled components knitted together by different firms for specific client needs. To understand the implications of this shift, we need to understand the shifts underway within the software industry itself. Today, software is everywhere—it is a highly pervasive sector that influences the characteristics of products, processes and services in almost every industry. The IT product industry today is over a trillion dollars in sales worldwide and is comprised of thousands of vendors developing products to meet diverse market needs. The hardware and software segments are complemented by a relatively recent segment composed of service (systems integration and consulting) companies. These companies help implement and integrate the disparate array of products that customers purchase. These service companies provide their services to suit specialized market needs such as product requirements gathering, design, development, customization, integration and maintenance. According to the NASSCOM-McKinsey Report [3], the global IT services market is anywhere

between $150 to $180 billion. Although there have been definite shifts in different delivery models dramatically affecting the fortunes of individual firms within the services market segment the overall market has been steadily growing over the years primarily fueled by factors such as Y2K, competitive pressures and global availability of expertise.

This whitepaper is intended to:

1. Serve as background paper to provide the basis for analyzing the various delivery models that IT service vendors can provide to their customers in an era of globalization of work;
2. Provide a framework or typology of work to understand the key factors that are shaping the architecture of global work.
3. Help in the design of global work processes by understanding the drivers of complexity.

THE GLOBAL SOFTWARE SERVICE ECOSYSTEM

In the early days of the IT industry we had a set of vertically integrated companies producing everything that a consumer needed (e,g., DEC, IBM and Wang). As described by Andy Grove [4], somewhere around the late 80's a transition from vertical integration to horizontal layers occurred. As a result of this transition, we moved from single firms offering end-to-end services to modular clusters [5] or stacks populated by specialist firms.

The industry stack divides activities into layers that are complementary to each other, as depicted in Figure 1. Today, just as was the case during the era of vertical integration, firms can deliver products that support most (if not all) layers of the stack. For example, consumers can buy chipsets, assembled computers, operating systems (AIX), middleware (Websphere), applications (CRM) or services (Global consulting) from IBM. The main difference in the era of stacks is that IBM provides these products with loose coupling and with open interfaces between them. As a result, consumers of these services have the option to mix and match IBM's products with those provided by other vendors. In the earlier era, this was not possible – a consumer had to pick a vendor and buy all required services from them. According to Lou Gerstner, former CEO of IBM, most companies specialize in one or a few layers and rely on other companies to offer complementary components [6]. Each of these components is layered above or below the other, and communicates through more or less standard interfaces, with closer layers being more related to each other than layers that are further apart in the stack. To deliver functionality to clients, we don't need hierarchies of vertically integrated corporation to coordinate production and distribution. Given the technical standards or design rules that exist between the layers, information costs of coordination have been greatly reduced. In fact, firms can tap low-cost producers anywhere in the world and coordinate through arm's-length exchanges between producers of standard components [7].

Lower layers and their components such as hardware and network services are often referred to as operating platforms and are fast becoming commodities. They have well defined interfaces with well defined terms of trade (prices). Firms build competencies on top of these lower layers by carefully selecting application packages and middleware packages and then launch business services on top of the application layer. The top layer contains all the service firms that provide services such as integration, training and maintenance. This layer includes consulting companies such as Accenture, IBM, Infosys and TCS. Middleware providers include firms such as IBM (with Webshpere), BEA systems (with Weblogic) and database vendors such as Oracle. In the application software layer one will find firms such as SAP, Peoplesoft (now part of Oracle), Siebel systems and others. In the business services layer companies that provide software as a service like Salesforce.com or Google's mapping program Google Earth.

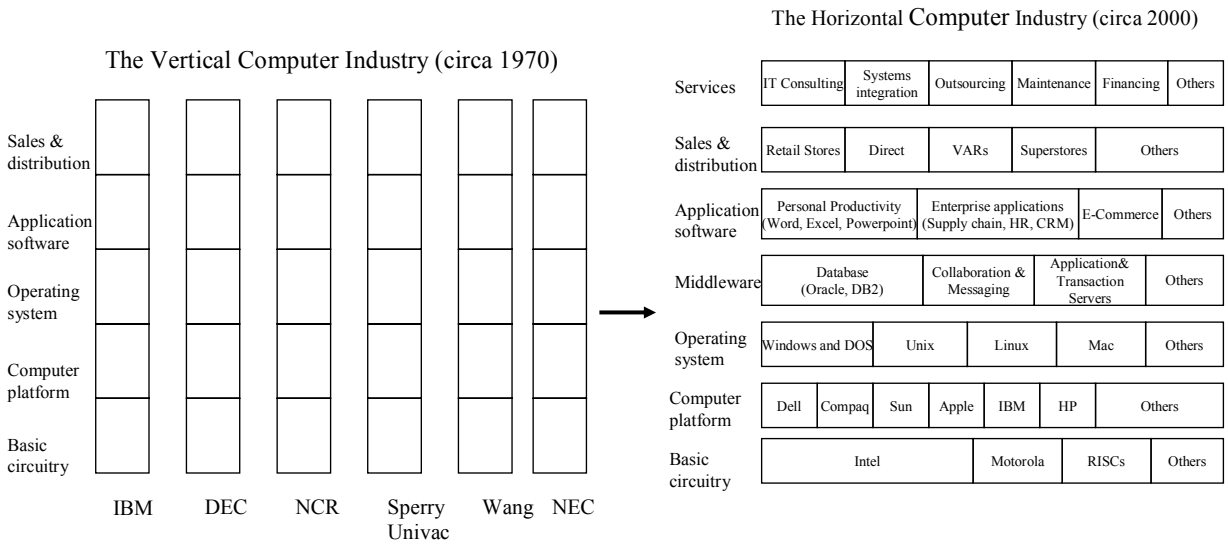

Figure 1: From vertical to horizontal transition

The software service market benefits from complementors – other firms offering products or services that increases your offerings' value to mutual customers [8]. Entities in this ecosystem include Fortune 500 firms that are customers of these services, vendors providing these services such as pure play service companies such as Wipro, Accenture, Keane, and TCS or product companies (like IBM) with service offerings, independent packaged software vendors like Oracle, SAP and Business Objects, advisory services firms like Gartner, TPI/Everest and IDC, and strategic consulting firms like Bain, McKinsey and BCG. In order to differentiate themselves from the competition, companies within this ecosystem form partnerships with one another and differentiate themselves by providing high quality services to their customers. These partnerships could be for reasons such as technology licensing, joint marketing or for getting their products and services to interoperate [9].

The stacks depicted in Figure 1 are generic to the IT industry. As one looks at the services industry, the different services offered by vendors can be grouped into layers that are often times referred to as the industry stack [10] that is depicted in Figure 2 shown below. The ecosystem described above is comprised of companies providing services within each one of these layers. The idea behind a stack is allow specialization within these layers while supporting interoperability across them. Companies focus on one or many layers of the stack and depend on others to provide services within the other areas. In general, from the perspective of the service providers, the lower layers of the stack have gotten commoditized and value migrates to the higher layers. As for the consumers of these services, they just want all the components from the various layers of the stack to work in harmony.

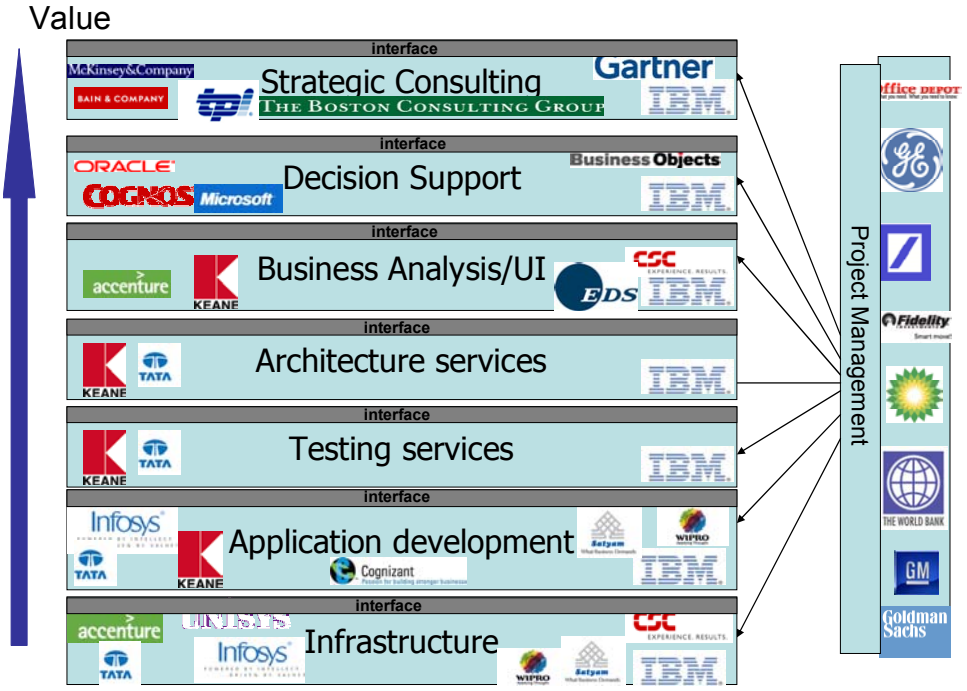

Figure 2: The Software Services Stack

1. Infrastructure Layer. This represents all the support structures that let enterprises connect, communicate and host applications. Within this layer, activities such as application maintenance, testing, computer and network management are done. In terms of components, this layer will contain hardware and networking devices, software like web servers, application servers and operating systems. Since infrastructure is made of components supplied by vendors that support well understood or open standards, it is becoming a commodity. Key players in this layer are Accenture, TCS, Infosys, Wipro, Satyam, CSC, HCL and IBM.

2. **Application development layer.** In this layer we have either off the shelf software or bespoke applications running on top of an infrastructure layer. Activities include writing programs to specifications from customer or installing

off-the-shelf software or writing software to get systems to interoperate. The scope for these projects could be a single application such as building a data warehouse for a financial services company or a set of applications for a healthcare enterprise. Companies that service this layer recruit and retain talented employees that understand the technologies and are programmers or certified project managers. Skills needed in this layer are things like .NET, J2EE, or the open source based Ruby on Rails platform knowledge or SAP, Oracle, Peoplesoft application package knowledge. Many vendors are involved in this layer. Some examples are Infosys, TCS, Wipro, Satyam, Keane, EDS, CSC, and Cognizant.

3. **Software testing**. When vendors develop software solutions for their clients, they have to make sure that the product is defect free and meets the functional and performance requirements outlined by the client. Defects and underperformance could harm their ability to meet service level agreements and affect their reputation. Vendors provide automated testing tools and services to improve the quality of the system and help meet the functional and performance requirements.

4. **Architecture Services.** When companies try to optimize their application portfolio in order to gain efficiencies in development, flexibility in response to changing market conditions or improve their ability to innovate using information technology, they launch an architecture analysis. At a high level architecture identifies the components of the systems and key design guidelines they ought to comply with [11, 12]. Architectural engagements can be about enterprise, applications or technical. The three vary in scope and intended target audience. Enterprise architecture includes the entire set of IT components and connects them to business value. Application architecture is targeted at the CIO and covers the functional features in an application. The technical architecture is about a particular implementation and the technologies that go into it. The purpose in doing it is to engage stakeholders in a dialog and align their strategy with that of the enterprise. Based on our research we found few companies to be operating in this layer. To operate in this layers vendors have to proficient in frameworks like Zachman's, The Open Group Architecture Framework (TOGAF) and Federal Enterprise Architecture Framework (FEAF). We identified IBM, Keane, and TCS as having strong practices for servicing this layer.

5. **Business Analysis Layer.** The next layer is about relevance to the business user of the system. To the end user, the user interface is the application. This has resulted in techniques such as user centered design that focuses on the system's users, their tasks, and their environment. Business analysts are responsible for analyzing the business needs of clients and stakeholders to help identify business problems and propose solutions. The business analyst is also the liaison between the user and the lower layers of the stack. Activities within this layer include requirements gathering, analysis and some prototyping. Key skills that an business analyst needs are proficiency in using Unified Modeling Language, methodologies like Unified Process and even some of the newer agile

development techniques like SCRUM are needed. Since activities in this layer involve deep interactions with clients to understand their requirements, interviewing and writing skills are critical.

6. **Decision support layer.** Once all the applications and databases are designed and delivered, decision makers would like to get reports to support their decisions. Senior management in companies, for example, track critical success factors – the vital few metrics that can successfully steer the enterprise through the environment. Information systems can be tuned to provide the data that create these metrics using software that supports this layer. Report writing and visualization tools are used to support stakeholders in this layer. Vendors in this layer are sometimes referred to as business intelligence vendors. Key product providers include Microsoft, Business Objects, Cognos, and Oracle. Being involved at this level allows a vendor to understand or be involved in the decisions at the lower levels. While product expertise is important, deep expertise into the client business is the differentiator. In one instance, we had a service vendor describe a project where they were managing the shelf space for a retail outlet. They started this project by helping manage the retail chain remote monitor their delivery trucks. This project grew in scope and now they are helping them manage their stores from a remote location in India.

7. **Strategic Business Consulting Layer**. This service layer helps companies improve their performance through rigorous analysis of current environment and business problems and co-create (with clients) plans for the future. Issues like changes to the current business model or identification of partners are dealt with by service providers in this layer. While few tools exist, many frameworks like value chain analysis and five forces analysis exist. Companies like Bain, McKinsey and BCG operate in this layer.

8. **Project Management**: This layer, a vertical one, represents all the competencies (skill, tools and techniques) that are required to successfully deliver projects. Almost all service organizations have this competency. In fact, some of them offer services like the project management office that will help clients run their projects successfully. Since project management can be provided as a service on top of any of the other layers, it is depicted as a vertical layer next to the customer.

Each layer has an interface that allows services from the other layers interoperate with it. This interface allows companies to specialize in bundles of layers and offer it as a meta-service. This can be seen in the case of IBM when they offer computing on demand. When purchasing, say CRM, from IBM, the customer does not have to worry about the infrastructure or application systems or the architecture. They simply buy the necessary amount of CRM services.

CLASSIFICATION OF GLOBAL WORK

The service industry stack described above provides a list of segments within which service vendors can operate. However, it doesn't provide a means to understand the different capabilities needed to operate within the layers nor the different challenges that confront the vendors and customers within each layer. A key challenge is the nature of workflow dependences between activities that are to be coordinated.

We follow a classic categorization of work defined by James Thompson [13] – an organizational sociologist. He classified workflow dependencies based on three types of interdependence: pooled, sequential and reciprocal. In **pooled interdependence**, work does not flow between organizational units. Each unit works independently and in parallel, following standard rules. The contributions of each unit are then aggregated to determine the total business activities. Standardization of the operational procedures in each unit enables them to operate without much communication or coordination. Examples are bank branches and franchises. In the software business (see Figure 2) we can see this happening when a company delivers packaged software with some customization. In this case, a company such as Accenture does some preliminary requirements gathering and recommends a product like SAP or Oracle. Once this recommendation is accepted by internal stakeholders, an internal budget and timeline is scoped out. This recommendation could sit with the client organization for a while, as they look for competitive bids from system integration vendors with that product expertise to do the implementation. Once a vendor, say Satyam, is picked they use the gathered requirements to do the implementation. IBM could be delivering the hardware, while Keane does the Architecture analysis. All these activities could proceed in parallel if standards exist for exchanging information. It is the client's responsibility to make sure that the project is successfully delivered. This may involve negotiating standard products or outputs from all the entities involved in the delivery of the project. Accenture may have to produce all the requirements as use cases, Satyam should produce the customization in a particular language and produce code documentation, and Keane must use Zachman's framework.

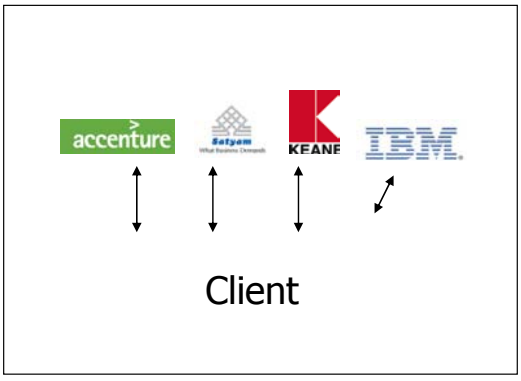

Figure 3a: Pooled interdependence

Sequential interdependence occurs in serial processes in which there are successive stages of production. Much focus is placed on planning the production sequence to maximize efficiency. The process is carefully monitored to ensure that the plans are followed. Examples are order processing and loan processing. This dependency type has medium communication needs and artifacts like plans, schedules are used to coordinate. In the service business, this happens when companies have selected the vendor to build a decision support system in a fixed time frame. In this case Accenture may start the requirements gathering and then bring in TCS (because of a pre-existing alliance) to do the development. TCS may recommend the use of Oracle for the backend and Business Objects to do the analytics. Each vendor from the ecosystem should follow pre-defined format or standards to produce outputs that the consumer of their output can use. In addition, if the companies agree and share on a common project plan, the coordination proceeds well. Similar to the previous dependence, standards must be negotiated. However, unlike the previous instance where the client owns and stores all the artifacts, clients just make sure that the dependent vendors agree on a standard for exchange.

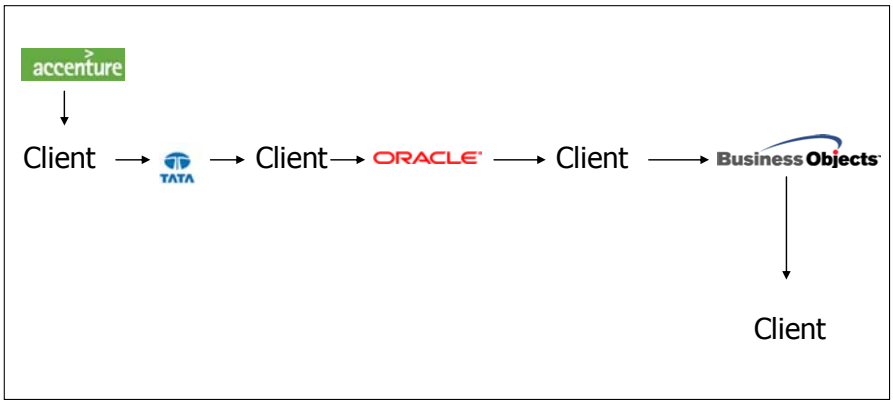

Figure 3b: Sequential interdependence

In **reciprocal interdependence**, various techniques and resources are employed to solve a problem without any predetermined ordering. The choice of technique to employ depends on feedback from the results of the other operations and is ad hoc. Because the operations depend on the give and take among participants, much interaction and communication is carried out to coordinate their activities. Examples are clinical health services and software engineering. This type of dependency happens in service project when clients are trying to build innovative applications. For example, the consulting company BCG may identify the need for a collaborative environment for a client and bring in Keane to do an assessment. Keane, because of a prior relationship with Microsoft, may recommend the use of the latest version of Sharepoint that has all the features needed to support the client. Since this is a new technology and a new initiative, many exchanges about product features, business requirements

and architecture take place between the four entities. To successfully deliver on projects, companies must agree on standards for both outputs and inputs.

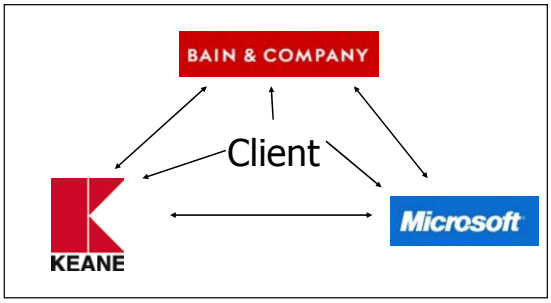

Figure 3c: Reciprocal interdependence

As described in the examples, each type of dependency calls for different coordination mechanisms. Clients should understand the nature of dependency and install the right coordination mechanisms. In some instances, the clients and vendors may not be on the same page with regards to the nature of dependencies and may result in mismatch of expectations, contracts and delivery.

TYPOLOGY OF GLOBAL WORK

As clients try to implement projects, they interact with the various entities in the ecosystem that provide specialized offerings around the various layers of the stack described previously. Since each layer of the stack is modular and independent in theory, many combinations of the stack layers can be offered as services such as business process outsourcing, remote services, staff augmentation and knowledge process outsourcing. These services have emerged based on vendor competencies and customer needs. In order to understand the evolution of the industry and the various services being provided we present the following framework that also depicts a typology of work.

On the x-axis, we identify the decision rights to deploy assets for the project. Assets could include resources and routines that allow the firms to efficiently transfer inputs to outputs. In our setting it is done using things like technology, process or people/knowledge. The question to ask is who determines the timing and extent to which these assets are deployed? The y-axis identifies the owner of the decision rights to adapt. This choice is important given the innovation imperative that organizations are facing today. When these innovations occur, ownership of the innovation could be contentious. In most outsourcing projects, the business decisions on adaptation are with the client. Today, some of the decisions about the client's customers are being made by the IT service provider.

As clients make their choices on the decision rights allocations, it results in the following four quadrants.

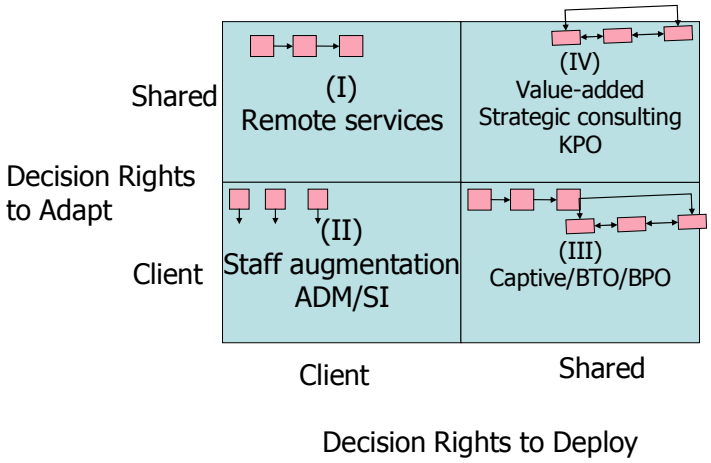

Figure 4: Typology of work framework

Quadrant 1: A recent trend in outsourcing has been the proliferation of remote monitoring services. As part of this value proposition, vendors manage the IT infrastructure of a client (say an insurance company). They use a control room staffed with experts from telecommunications, application server software, web server software and in some cases the applications themselves. While the client owns the right to change and develop the infrastructure itself, the vendor determines the best way to track performance and implement process to monitor the infrastructure.

While this was in and by it self novel, what makes it interesting was that the vendor facility could be in Chennai, India and the insurance company could be located in North America. The client owns the infrastructure of computers, networks, servers and applications. The vendor, using tools like HP's OpenView or IBM's Tivoli software, can monitor the insurance company's infrastructure from several thousand miles. When the client encounters a problem with the infrastructure, it could be analyzed first by the vendor and the solution strategy implemented and fixed using the insurance company's own IT group. Since the vendor typically executes a contract containing service level agreements, they are able to make decisions independent of inputs from the client.

Service vendors operating in this layer may not have deep hardware expertise. They use the system management software created by other vendors to identify, triage and isolate problems. The problem could be with a hardware system or the infrastructure software like the web server or an application server. The vendor can then use either a hardware vendor or the service department of the software vendor or in most cases the IT department of the client company to fulfill their commitments.

Since the decision rights are mostly owned by the client, the dependency between the client and the vendor is mostly sequential. Coordination can be performed by plan. This means that there must be *a priori* agreement on the schedule of the workflow. When the workflow needs to be adapted to deal with a new type of problem, the vendor can change the existing workflow without seeking permission from the client. For example, if the vendor has a new operations center in China and would like to route some of the monitoring to that country, they can do that without seeking permission from the client.

Quadrant 2: This business model, staff augmentation, has been around for a long time. Under this model, a vendor would supply the client with resources that are directly managed by the client. For example, a financial services client may need a DBA for the next six months and would make a request to the vendor. Since they have a relationship with an IT vendor, the vendor makes sure that they get the required resource with the right level of experience and skill set. The vendor is able to do this because they have a deep bench of resources that they share across clients. The client benefits because they do not have to employ and train resources during periods of unpredictable demand for projects and skills.

How the supplied resources are utilized in a project is completely up to the client. The assets deployed and used in the project are also done based on the clients needs and by their managers. In addition, when project requirements change the client decides how and what to adapt processes to meet requirements.

The core layers of the model that are supported by the vendors servicing this quadrant are the applications layer and above. Given the dependency between the vendor and client is pooled, this quadrant created the least coordination burden. Coordination is achieved by standardization of tasks. This is possible when activities are specifiable, measurable and predictable [14]. As a result, the client organization must insist on routinization of tasks that result in internally consistent set of rules.

Quadrant 3: The ground breakers in the IT outsourcing industry like Amex, British Airways and GE started with this business model by opening captive centers in India. These captives were dedicated centers that would process back-office work for these companies. In the case of Amex it was finance functions for credit cards serving Asia-Pacific, Japan and Australia [15]. In 1996 British Airways, through its WNS subsidiary, pushed all the back-office work related to customer service management to India. GE opened the first call center for dealing with un-matched checks received for credit card payments [15]. Under this model, the customer provides the business logic and rules for decision making that the vendor codifies into the system. In addition, the vendor could help the client with setting-up the centers, hiring staff and putting in place methodologies. Once the center is up and running, the client could decide to run it themselves or have the vendor run it for a period of time.

Looking at it from the perspective of a stack, the infrastructure and assets are initially owned by the vendor. The vendor builds the capabilities all the way up to the application layer. When the business logic has to changed and baked into the application layer, the vendor makes those changes with the permission of the client. In some instances, in order to meet client requirements, the vendor may make investments in the infrastructure and share it with other projects. Sans any prior agreements, the client makes decisions about all project related issues.

This quadrant, given shared ownership of capabilities, creates mostly sequential and pooled dependencies and coordination is performed in a manner similar to quadrants 1 and 2.

Quadrant 4: In this quadrant the type of problem solving is mostly unstructured. This mandates the vendor to move away from rule-based decision making around knowledge work to more judgment-based problem solving activities. The assets that are involved in this quadrant are more knowledge based like business methods or process methods. When developing new products, participants may combine components, services and intellectual property from many sources and these may not even be defined in advance. They are designed and created through a process of simultaneous/concurrent engineering. In our context, a vendor may work on a project with a client that involves doing statistical analysis of clinical trial data. Initially the vendor could simply run the requested statistical models and provided the results to the client (this would be rule based and fit quadrant 3). To move to quadrant 4 they could hire doctors and other econometric modeling experts who can pick the right statistical models and interpret the data for the clients.

To operate in this quadrant, the vendor should treat layers below the decision support and consulting as commodities or as a bundled service. They differentiate themselves based on domain expertise such as financial services, retailing etc.

Given the unstructured nature of problem solving in this quadrant, reciprocal dependencies exist between the client and vendor. Coordination is done my mutual adjustment. This type of coordination involved communication across peers or along horizontal lines. This coordination is best when the situation is ad-hoc and uncertain. Once the problem is solved, the dependencies revert back to their sequential nature. Information systems can facilitate the co-creation of value by accelerating the formalization of tacit knowledge and facilitating transparency [7].

As service providers begin to use this framework they have to consider a few issues. Clients may have some assumptions about the relationships that are either directed towards squeezing efficiencies or developing innovative solutions to their business problems. The relationship governance models have to be aligned with those expectations. Executives pointed out that in some instances the governance model in place is for the BPO (quadrant 3) while the execution model

is mostly staff augmentation. These types of changes make happen over the course of the project as changes occur to the scope or the environment.

Key Factors in Effective Global Work

Our research has identified the following to be key factors from the perspective of the vendors: nature of relationship with client, key client sponsor, ownership of IP generated, critical capability, revenue model, innovation and metrics tracked.

Quadrant 1: The vendor providing services typically interacts with the CIO of the client organization and the main concern is total cost of ownership. All the assets are owned by the client and the vendor is paid based on volume of service/transactions processed or provided. Typical metrics tracked are infrastructure uptime and time to respond to queries. Any innovation to the service that is made as part of the engagement is owned by the vendor.

Quadrant 2: As mentioned earlier, this was the starting point for the industry. The vendor interacts closely with the project manager of the client organization and plays the role of a supplier. The intellectual property that is created by the vendor's resources is owned by the client. The vendor has a transactional relationship and gets compensated based on the number of people they provide. The resources provided by the vendor are managed by the project manager. As we look at the history of the industry, we find that while playing the role of an individual contributor, these vendor resources picked up domain skills like financial services, healthcare or manufacturing. This allowed many vendors to move to quadrant 3. Innovations made in this quadrant, unless otherwise specified in the contract, are owned by the vendor.

Quadrant 3: The vendor now has a business unit sponsor within the client organization and plays the role of a solutions partner to the business unit. The main asset they provide is domain expertise in addition to the development skills. An IP that is created is owned by the client, unless otherwise contractually stipulated. Relationship is managed using service level agreements.

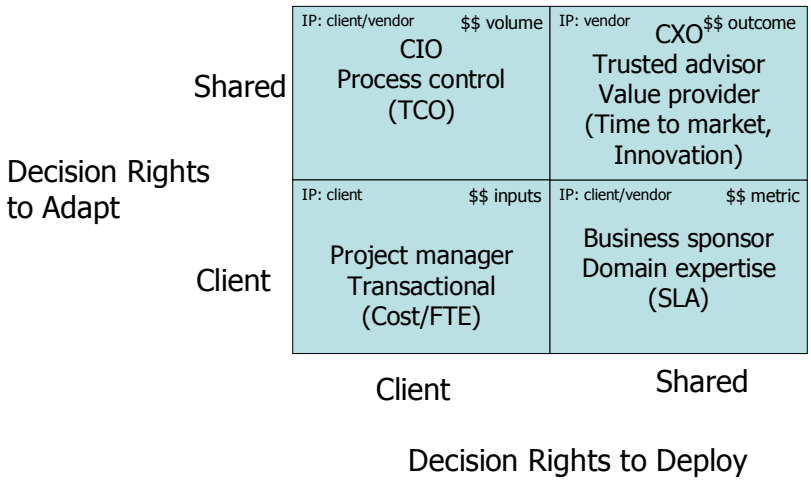

Figure 5: Key Factors within each quadrant

Quadrant 4: This is a potential high margin business. Typically revenue per employee goes up while operating in this quadrant. As a trusted advisor, the vendor provides strategic advice on commissioning and decommissioning projects. Vendors begin to interact with the board and senior level executives advising them on the type of projects they should under and type of contracts (fixed price, time and materials). They could also help with determining if staff augmentation or BPO or KPO is the ideal model. Outcome based metrics like time to market or innovations or sales growth could be used to compensate the vendor. As many IT vendors aim to reach billions of dollars in revenues, their employee base grows proportionately. In is true that they can get some productivity using knowledge and code reuse systems but the high multiples come from moving into higher margin business like those seen in quadrant 4. However, in order to win here speed and responsiveness are vital. This, in turn, is dependent on shared social context and language to ensure shared understanding between partners.

As we reflect on the four quadrants, it is quite obvious that globalization has had an impact on all of them. In particular, the globally distributed nature of expertise, the resulting global distribution of work, and customer demands for 24x7 operational availability have brought with it its own challenges. The coordination burden, however, is different for each one of them. Vendors can profitably operate in any one of the quadrants. In fact, some vendors operate in multiple quadrants. Some of quadrants like 3 and 4 are relatively new and we have seen many new entrants there. The key point is for customers and vendors in the relationship to recognize the operating model and put in different capabilities (technological and process oriented) and governance regimes to fit that model.

Conclusions

Most IT vendors have plans to generate around $10 billion dollars in revenue by 2010. This is challenging because the service business is very people dependent and sourcing of talent has become very critical [16]. As revenues double, the staffing requirements double with it or in some cases, given the overheads for coordination, more than double! What should vendor do in order to increase top line while keeping costs down? The other challenge is a byproduct of the global nature of the workforce. Given the growing complexity of IT projects, distributed nature of expertise and global footprints of clients and service providers, we have identified key factors for successful delivery of information systems. Our recommendations are grouped into three categories: project management, process management and technology.

Project management

Global collaboration expertise: As companies move from a core/periphery model of delivering work to a global peer-to-peer model, the scarce competitive resource is their ability to manage complex relationships and teamwork across cultural and linguistic barriers[7]. Quadrant 4, where this needed the most is also where the high margin projects exist. As described earlier, this quadrant is needs to support reciprocal dependencies between member of the ecosystem and places a high demand on collaboration capability. As firms try to deal with the globalization forces described earlier, collaboration capabilities become vital. A project that gets initiated by one entity gets divided up into multiple chunks, sourced to other entities in different locations and recombined at the very last minute and delivered to the originator of the workflow. This type of project work delivery can be found in settings such as software development, claims processing, help desk support and legal services. While there are strategic and economic challenges facing this type of work, a critical success factor is the coordination and monitoring of the workflows involved in completing this work. In particular, in knowledge-based activities like IT services, the ability to share and synthesize knowledge and coordinate actions is critical to effective performance. In almost all instances, there exists a technology platform behind successful delivery of distributed work. What capabilities should be embedded in these technology platforms that support the coordination and successful delivery of such projects?

Soft skills: workforce on-boarding process is essential for global projects involving employees from several countries and cultures. Establishing shared norms are important for working with staff from many vendors that are part of the ecosystem helping deliver on the client project.

Listening to the edge: Since many projects involve people from several geographies and vendors, it is very difficult to centralize information flows. Just as Google does with its employees, it is important to have project information available and transparent to the entire organization (except when compliance

requirements prohibit it). Google does this by having employees maintain their websites and blogs and allowing other colleagues to access it. This allows for project related adjustments, knowledge sharing for problem solving and intelligence gathering about the clients.

Technology management

Alliance ecosystem management: Service firms form deep relationships with software product vendors. In one instance, a service vendor had such a deep relationship with an ERP vendor that their consulting clients would look to them to get some changes incorporated into the ERP package. The ERP vendor benefits because of the fact that these vendors help them sell more licenses. The ERP vendor also recommends the service firm to its clients. In fact, when new features are added to the product the service vendor is already aware and trained to serve the customers along with the release of the product.

As these alliances are formed, they create business ecosystem that have emerged as the new referent for strategy formation. Hence, it is very important for IT vendors to map the ecosystem in terms of inter-firm interconnections. This helps them understand how competitors access complementary resources through relationships. A diagrammatic representation of the ecosystem with key relationships provides the context for strategy formation and implementation.

Architecture capability: As firms are trying to move into higher layers of the stack and perform value-added services, it is becoming important to understand the architecture of the client's applications and their business models. This way, they can get more involved in discussions around aligning IT strategy with business strategy [17] and recommending what systems get built as opposed to simply building them. As a result they can be involved in the early stages of the project starting with the RFP calls.

Another reason for working in this layer is the client's competitive environment. Given the high intensity of competition in many sectors, clients are thinking about their own organization in terms of functional competencies or services and how they orchestrate these services to deliver value to their clients. Documenting the current architecture helps them understand the current dependencies. When firms want to address new market needs by deploying these services in new ways, a concept termed as architectural innovation[18] or combinative capability [19], they need to be effective in reconfiguration. This, once again, requires architectural understanding.

Domain expertise: traditionally, IT vendors have developed skills and collateral around technologies like Microsoft stack, Oracle DBMS, J2EE, .NET, or the Open Source stack. Now they are expected to provide advice within domains like financial services, manufacture and healthcare. They are expected to have deep understanding of compliance requirements, best practices and industry trends. While many of the skills needed for successful IT projects may get commoditized,

domain expertise keeps growing and changing. In fact, this allows a vendor to provide innovative ideas to the clients based on their expertise in the same domain or something borrowed for a different domain.

Process management

Methodologies: As we consider the software development life cycle, at the very beginning the dominant activity is requirements gathering. During the phase an analyst talks to business users and documents requirements. This document is later passed on the design and then to the development team. This process works well when separation of concerns and specialization is the primary concern. However, requirement gathering is not a one-shot process. In many instances, it has to be done in an iterative fashion.

New methodologies for development like Agile techniques are becoming critical for client satisfaction and delivery. Old waterfall techniques are good when the requirements are well understood and the client will learn to live with the system that they get. Today, using some of the agile development techniques, analysis and development takes place in an iterative and collaborative fashion. This couple with open source software development and frameworks like Ruby on Rails are forcing companies to train a new cadre of specialists.

Reuse of knowledge and code: With many vendors simultaneously handling several hundred projects, teams tend to work in silos with very little sharing of knowledge or code. One vendor has developed an extensive system to share software components across projects. In this company they have provided incentives for project managers and team members that are based on others using components created during their project delivery. Another vendor has created a knowledge management portal that is based on social tagging technology. This system allows contributions to be tagged based on the contributors' tags. While these tags are created within silos, similar tags cold be linked using social software tools and presented as relevant links to users as tag clouds or useful links.

User experience design: Most vendors have developed capabilities to deliver useful functionality. The ones that are good at doing usability analysis and user centered design are separating themselves from the crowd. Companies are using scenario planning and workflow analysis to understand user context and then designing application to fit that. Techniques like user-centered design help provide users with applications that support their work without much technical training on how to use the information system.

Global Recruiting: Given the high turnover in this industry, an organizations ability to find and retain talent globally is critical. Many firms are focused on college recruiting. One firm hires several hundreds at a time and sends them through training programs. These programs allow them to pre-qualify a recruit into their roles such as business analyst or developer. From a global perspective,

firms are developing capabilities to acclimatize and manage global teams. As teams prepare to work on global projects, orientation to the new environment is provided by experts on culture and team work. This helps team members to be productive contributors to global work.

Reference:

1. Friedman, T., *The World Is Flat: A Brief History of the Twenty-first Century*. 2005, New York: Farrar Straus Giroux.
2. Campbell-Kelly, M., *From Airline Reservation to Sonic the Hedgehog: A History of the Software Industry*. 2003, Cambridge, MA: The MIT Press. 372.
3. Sinha, J., K. Karnik, and N. Kaka, *NASSCOM-McKinsey Report 2005: Extending India's Leadership of the Global IT and BPO Industries*. 2005, NASSCOM-McKinsey: New Delhi.
4. Grove, A.S., *Only the paranoid survive: how to exploit the crisis points that challenge every company and career*. 1st ed. 1996, New York: Currency Doubleday. xii, 210 p.
5. Baldwin, C.Y. and K.B. Clark, *Design Rules: The Power of Modularity*. 2000, Cambridge, Mass.: MIT Press. v. <1 >.
6. Gerstner_Jr, L.V., *Who Says Elephants Can't Dance? Inside IBM's Historic Turnaround*. 2003, New York: Harper Business;. 384.
7. Saxenian, A., *The New Argonauts: Regional Advantage in a Global Economy*. 2006, Cambridge, MA: Harvard University Press.
8. Yoffie, D. and M. Kwak, *With Friends Like These: The Art of Managing Complementors* Harvard Business Review, 2006. **84**(8): p. 88-98.
9. Iyer, B., C.-H. Lee, and N. Venkatraman, *Managing in a "Small World Ecosystem": Some Lessons from the Software Sector.* California Management Review, 2006. **48**(3): p. 28-47.
10. Gao, L. and B. Iyer, *Using Software Stacks to Explain Complementarities: The Case of Mergers and Acquisitions in the Software Industry. (forthcoming).* Journal of Management Information Systems, 2006. **23**(2): p. 121-149.
11. Baldwin, C. and K. Clark, *Design Rules: The Power of Modularity*. 2000, Cambridge, MA: MIT Press.
12. Iyer, B. and R. Gottlieb, *The Four-Domain Architecture: An approach to support enterprise architecture design.* IBM Systems Journal, 2004. **43**(3): p. 1-10.
13. Thompson, J., *Organizations in Action*. 1967, New York: McGraw-Hill.
14. Christensen, C. and M. Raynor, *The Innovator's Solution: Creating and Sustaining Successful Growth* 2003, Boston, MA: Harvard Business School Press
15. Parthasarathy, S., *The Backroom Brigade*. 2006, London: Penguin Books.
16. Wooldridge, A., *The battle for brainpower*, in *The Economist*. 2006.
17. Henderson, J.C. and N. Venkatraman, *Strategic alignment: Leveraging information technology for transforming organizations.* IBM Systems Journal, 1993. **32**(1): p. 4.
18. Henderson, R. and K. Clark, *Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms.* Administrative Science Quarterly, 1990. **35**: p. 9-30.
19. Kogut, B. and U. Zander, *Knowledge of the firm, combinative capabilities, and the replication of technology.* Organization Science, 1992. **3**(3): p. 383-397.