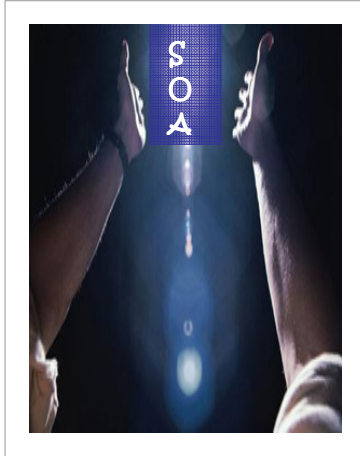


How to create value from Service Oriented Architecture

Sukhendu Pal and Jon Fuller

July 2006



The push for the Service Oriented Architecture (SOA) by technology companies has the potential to deliver cost-effective and agile information systems. But can SOA make an enterprise agile without an organisation dedicating itself to provide superior services to its customers? It is highly doubtful. The transition to an agile enterprise to compete in today's volatile business environment will require a change in CEOs mindset first. Today's changeable markets require companies to value flexibility and speed over the obsession of efficiency and scale. In an unpredictable world of business, it is not the biggest companies that win. Rather it is the most adaptable companies, equipped with the best technology, smart people, and superior services that prosper. Find out how CEOs can guide CIOs and CTOs to build agile enterprises and create sustainable value by taking advantage of SOAs.

While business has entered the information age, the operating model and supporting information technology (IT) architectures of many companies is still rooted in the industrial age. Many companies find it hard to get to market quickly with new products because their complex and inflexible IT architectures force them to develop many parts of each new product almost from scratch. Reuse of existing components is rare. The quickest way to get out new offerings is simply to patch the existing architecture by forging connections immediately between systems. The result is an increasingly complex, convoluted architecture littered with many servers, incompatible stand-alone applications, operating on software from a large number of technology vendors. In the early 1980s, many technology vendors pushed the idea of "departmental computing" and flogged their infrastructure software to increase their revenue while increasing complexity in clients' organisations. Today, these same technology vendors are promoting SOA as the building blocks for flexibility. No wonder CEOs and their CIOs are sceptical of these technology vendors' motives.

Service Oriented Organisation (SOO) is a prerequisite to a SOA

The critical question facing most CEOs is: how does a company become agile without forfeiting its cost and scale advantages? In the conventional "make-and-sell" business model, variety is the enemy of scale^[1]. Today, customers demand variety, speed and superior service, but this comes at a cost. So, how can a company become agile without compromising its scale and cost structure? The key is modularity of products, IT architectures and applications, and decoupling of business processes and value chains (see *Why integrate, when you can aggregate?* by Pal and Hammond). But even more important factor is the senior management's focus on outstanding customer service. Most CEOs know it is easy to deliver lousy service. But consider this: How about a business world where organisations treat customers better and it benefits them as well? Such smart organisations exist, but unfortunately not many. Their customers are better off and the organisations earn a disproportionate share of the profits over their competitors. Before embarking on a new technology venture, these smart organisations use three successful strategies to differentiate on service and they do it in a sustainable manner (see Figure 1).

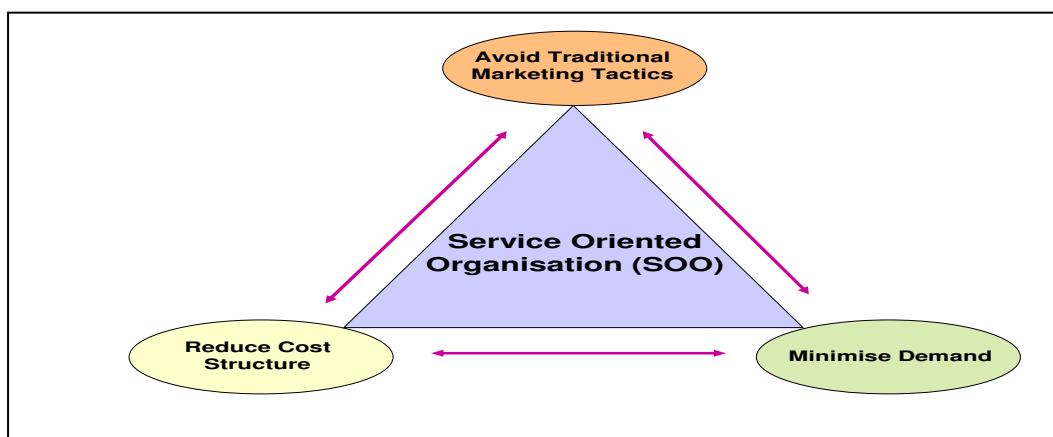


Figure 1: Strategic imperatives for a Service Oriented Organisation (SOO)

Creating a Service Oriented Organisation (SOO) that is capable of anticipating customers' needs and responding quickly with the appropriate capabilities and offerings is a transformation that needs to begin at the top with the leadership of the company. A service oriented organisation is a sign of an agile company, and it is a pre-requisite for SOA that delivers value. Complex IT infrastructure and challenge of implementing SOA is not the biggest hurdle. A much difficult hurdle to overcome is the product-centric mind-set of most senior executives – because their compensation is based on the P&L structure that perpetuates that approach. Product manufacturing or "make-and-sell" business model remains the basic building blocks of the P&L for most companies – from banking to insurance,

from pharmaceuticals to retailers, from utilities to mobile operators, from software vendors to hardware vendors – and the cost structure of a product business raises all kinds of barriers to service-oriented organisational investments.

Companies struggle to offer customers new products and services at speed and prices with features they want. The "customer experience" is frustrating and often on an unstoppable downward trend. The problem is old thinking of mass production, carried out by product-centric companies. For all the talk of focusing on the customer, the total experience of the consumer is ignored. A good example of these strategic imperatives not working in a spectacular way is the mobile telecommunication industry. It sees the customer as an adversary, (asking the customer to sign contracts for the next eighteen months to two years) so they don't have to take care of the customer. It is like they don't trust themselves to deliver good service. For their long-term service contracts, customers have to guess their usage. The companies make money when customers guess wrong (e.g., when customers pay for capacity they don't use). The mobile telecommunication companies have set up the customers in an adversarial role; these mobile companies benefit when customers do not. For their part, mobile telecommunication companies complain their customers are price-sensitive and will not pay for service. But the mobile telecommunication industry is like the banking and insurance industries where customers are often misled for the sole benefit of banks and insurers. For example, complaints against the UK high street banks rose by almost half in 2005 amid growing signs of customer dissatisfaction with the sector, which has reported record profits. The Banking Code Standards Board (BCSB), which operates a voluntary code of conduct for the industry, said it had received 3,500 complaints and enquiries in 2005 – up by 50 percent since 2004. As complaints have surged so have the banks' profits. In 2005, the total profits for the sector topped £33 billion. This means the Big Five UK banks — HSBC, RBS, HBOS, Barclays, and Lloyds TSB — made at least £4 million an hour between them while customer complaints grew substantially ^[2]. The question is: how long can these banks, for example, continue to grow in the UK with dissatisfied customers?

Avoid traditional marketing tactics

For example, some UK banks are trapping customers in accounts that promise eye-catching rates (such as 8 percent annual interest rate) and easy access, but penalise customers who wish to invest more than £250 per month or make withdrawals. These banks argue that they make the terms clear, but customers are caught out by "confusion marketing". Bank accounts have become more complex making it difficult for customers to compare offerings. Banks are making a killing from confusion marketing.

But can this "confusion marketing" help banks win the majority-share of customers' wallets? It is doubtful. In today's customer driven world, value flows horizontally through a company in the form of service chains pulled through by the customer. This is the reverse of companies trying to "make-and-sell" products by way of integration via vertically organised departmental functions or silos. Today, complex products with "confusion marketing" do not bring prizes. Instead, services that retain customers' loyalty allow companies to win the heart, mind and majority-share of customers' wallets. The choice for companies is simple: either a company continues to flog products in the canny way and gradually experience loss of market-share, or become a service oriented organisation that dominates its chosen markets, and keeps the competitions locked-out.

Manage cost structure

In the last five years, many CEOs have taken the path of offshoring to address the increasing complexity of their IT and escalating cost of running IT organisations. By moving some business functions to countries with lower wages and by leveraging offshore service providers with scale economies, a handful of companies have achieved reductions in their cost structures. However, many of these companies run into problems trying to segregate offshored applications from tightly coupled surrounding systems located onshore. The problem with such approaches is that while companies can lower the IT cost coefficient, they rarely change the cost equation. As internal and external IT systems continue to grow in complexity, many CEOs will soon find themselves back where they began. Changing that cost equation with certainty requires that IT architectures are re-engineered to enable systems to grow increasingly responsive and cost-effective over a period of time (see Figure 2).

Today, most CIOs find themselves in a vicious cycle – every year the run cost for keeping the existing systems functioning takes a bigger and bigger share of the IT budget. That means more money is spent on keeping the "lights on" and less money on new development. Less money spent on new development means that changes made tend to be strapped onto an already inadequate system rather than in a scaleable and agile IT architecture. This way of working increases complexity in the IT systems, that in turn leads to more money in keeping the "lights on".

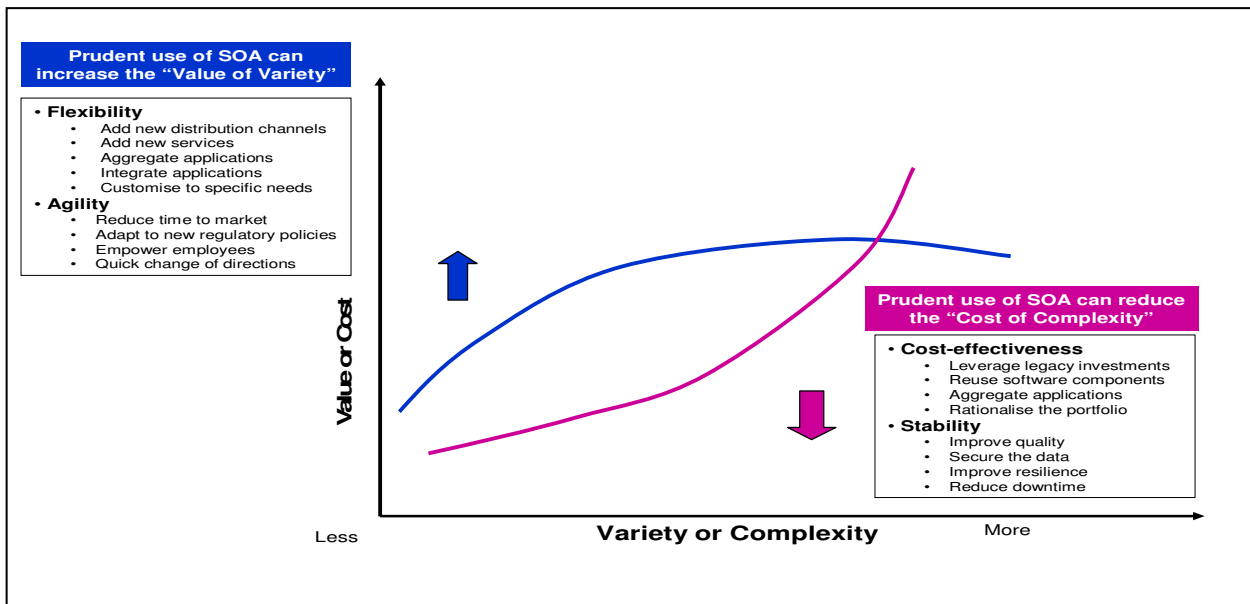


Figure 2: Cost and Complexity trade-off

Minimise demand

The service orientated organisations are geared toward minimising the need for customer service. So their strategy is to minimise demand. However, many other organisations have the strategy of hiding the phone number of customer services so deeply on their websites that customers simply cannot call them – that is their way of minimising demand.

At service orientated organisations, customer services are excellent and the need for it is rare. Employees answering the phone are part of the product development team and they are expected to talk with the technical experts about the calls they receive. These employees are paid more than their counterparts in other companies offshore call centres.

What exactly is SOA?

SOA is a component model that inter-relates the different functional units of an application, called *services*, through well-defined interfaces between these services. SOA can be seen as a collection of patterns for building an organisation level aggregation layer between applications while abstracting those applications as services. It is a way of simplifying technology through the use of components or building blocks that can be used and reused throughout the organisation and can communicate with each other. It means, for example, that data should never have to be entered more than once.

Where to start

Our work with clients over the last five years sheds further light on how service orientated companies use SOA to bring agility and accelerate business growth. The successful companies couple the design of the business strategy with design of SOA based IT systems (see Figure 3).

Increasingly many smart CIOs are self-funding their IT investments in SOA using a layer of abstraction to legacy system deploying a technology platform, such as SAFE™, that simplifies aggregation of data, servers, applications, and processes. These CIOs find they are able to use SAFE™ to extend the life of legacy systems by bundling applications into services and simplifying the expensive integration with other systems^[3]. The new business process driven architectural simplification not only accelerates the IT system's ability to incorporate new changes, it can also do so while increasing system resilience and lowering costs.

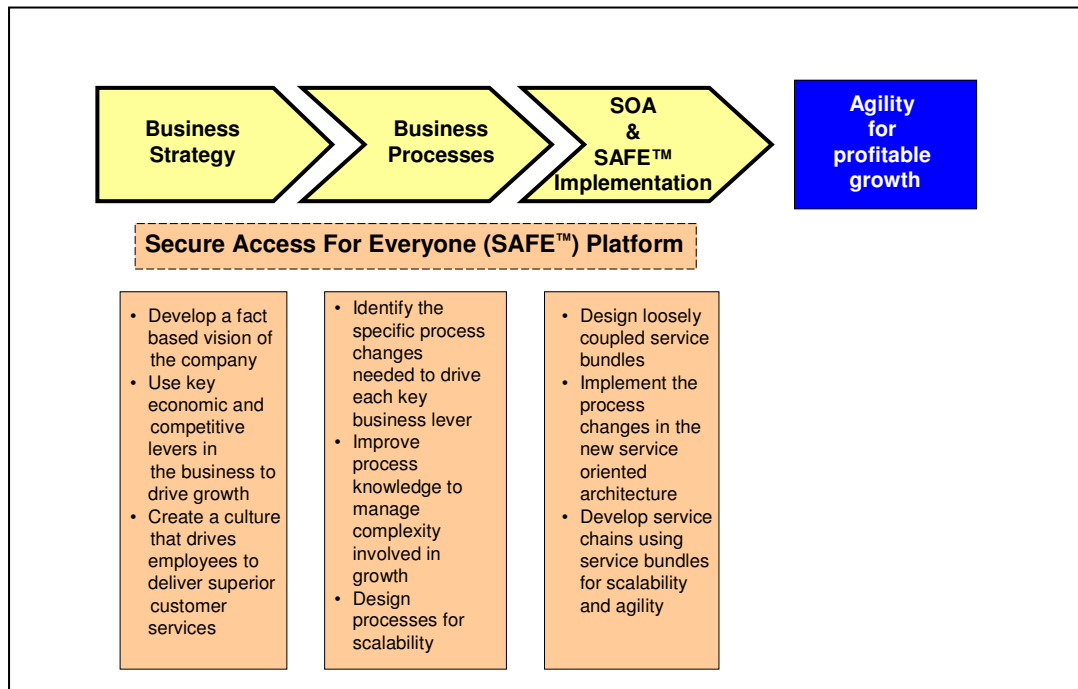


Figure 3: The SOA framework to achieve organisational agility for profitable growth

Developing a SOA migration strategy

Migrating to an SOA is not a technology project as many technology suppliers claim. It is a business initiative. For example, what services does a business need to run the processes? Where are these services going to come from? What happens if a service goes down? How long would it take the business to recover? These are questions that the average IT person is hardly qualified to answer. Developing a SOA will therefore require close cooperation between IT departments and business units – traditionally a challenging proposition in most organisations.

Many large organisations providing multiple services find that their IT has been heavily departmentalised to allow each business unit to manage its own IT needs, instead of relying on a centrally managed IT organisation. As a result, many business units ended up creating applications running on their own servers in isolation from other business units. Consider a major mobile phone operator - it will have several billing, customer relationship management, enterprise resource planning systems supporting different aspects of its business. The company installed many "best of breed" applications without considering integration with other applications. They later find that special arrangements are often required for making a new application a part of the IT infrastructure (see Figure 4).

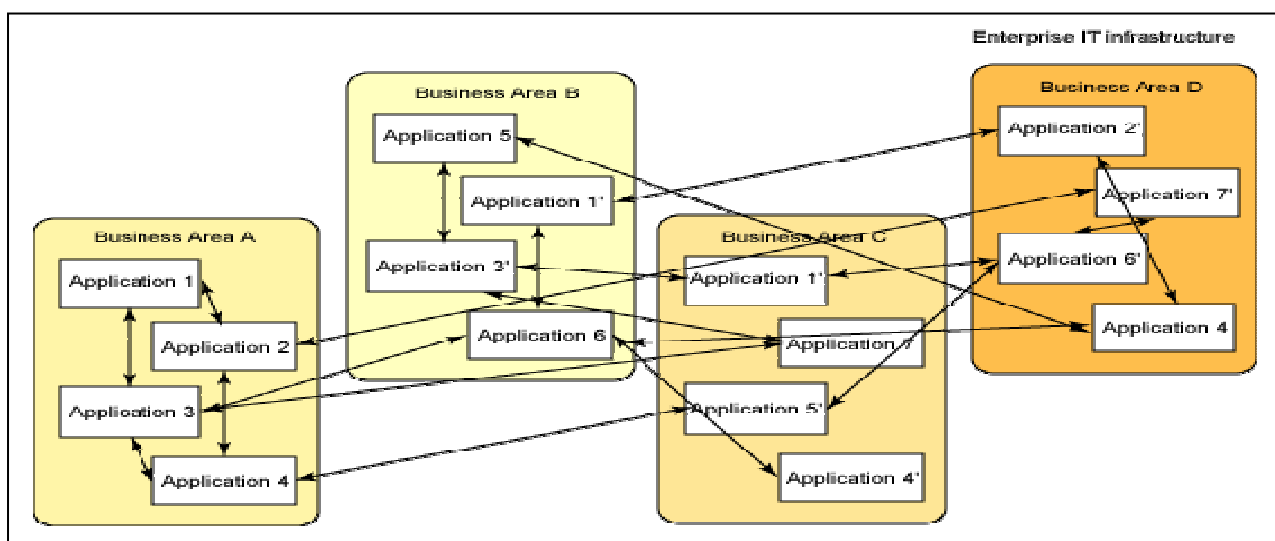


Figure 4: Example legacy enterprise IT infrastructure of a mobile operator

The above diagram shows a method employed by many companies, where an application with the required functionality is deployed in several business areas (for example, Application 1 and Application 1'). Often, the application is slightly modified with each deployment to meet unique requirements of a specific business area. While there are no obvious disadvantages to having multiple applications with the same functionality, their existence indicate the following:-

- Potential existence of data duplication in the company, which compromises data integrity;
- Higher cost of maintaining multiple applications;
- Special care needed when such applications are consolidated to reduce impact on applications that have been interdependent.
- These problems can be addressed by applying SOA principles for a successful integration of applications on the enterprise level. Splitting a complex and departmentalised enterprise into a number of separate segments is likely to make SOA implementation much easier (see Figure 5).

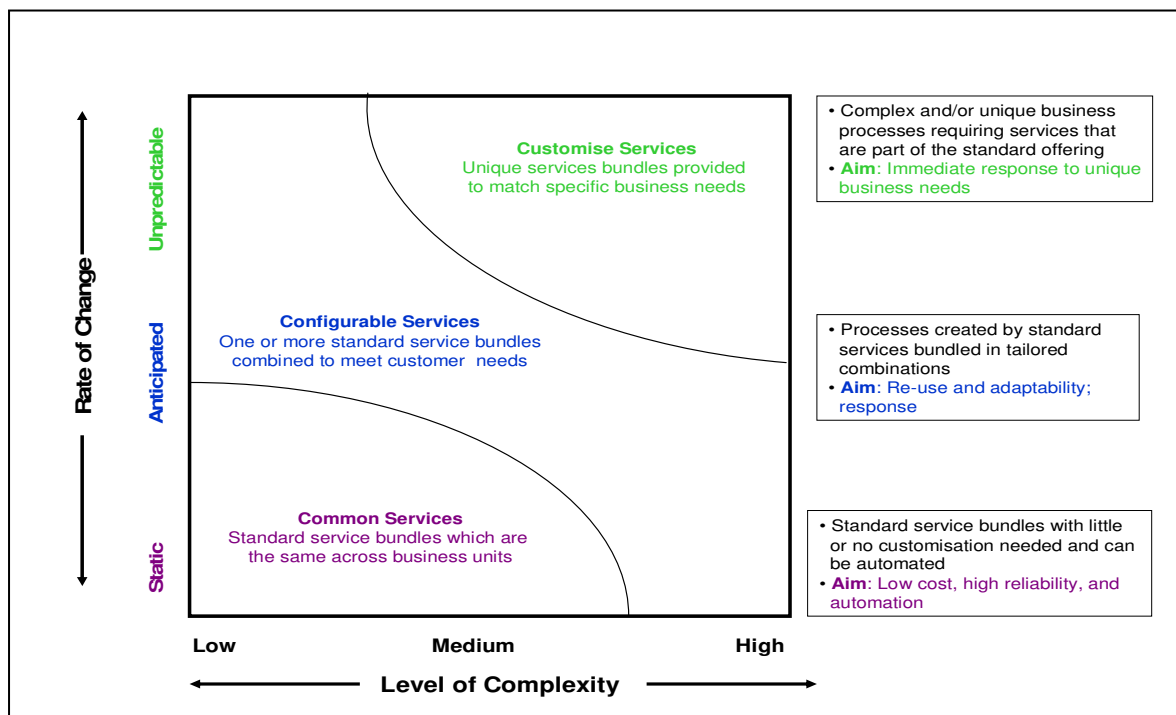


Figure 5: Segment services by complexity and rate of change

Define service bundles for separate domains

Deciding what constitutes a service is always a challenge – include too little functionality and a service is unlikely to be useful. On the other hand, include too much functionality and a service may be unusable outside the context for which it was originally written. The following criteria may be used for determining if a set of services belong to a separate domain:

- *Functional domains* are selected based on a business function. Services in these domains consume a limited number of services from the outside of the domain and expose a limited number of services to the outer enterprise.
- *Technology-based domains* are selected based on a set of technologies they utilise. These may include mainframe applications, distributed applications, etc.
- *Application-based domains* are tightly coupled collections of applications. A collection of applications that share the same database may be an example.

Once services are identified, domains need to be separated from each other by establishing clear boundaries enforced through use of gateways and firewalls. Such separation allows for better control over application interactions and further flexibility for making changes to applications without affecting the rest of the enterprise. Such separation may be achieved through several different approaches:-

- Defining *business functions* that expose functionality of a domain as a coarse grained service: Business functions are driven by needs of the enterprise, rather than internal implementation details. Business functions may call finer-grained technical services within the domain internally, providing a service to external users. The main objective of defining business functions is limiting interactions between domains to well-defined points.
- Adding *gateways* that translate assumptions of one domain into assumptions of another domain: Gateways also allow establishing clear separation between external entities (business partners, for example) adding better control capabilities. Often, gateways may be combined with business functions into a single logical entity. However, adding a gateway is dictated by business requirements and may not be needed if domains share the same assumptions.
- Gateways could be implemented using one of the following approaches:-
 - *Transparent/Proxy Gateways* expose domain services in such a way that a user believes that it interacts with a business function directly. Such gateways may also perform transformation of incoming and outgoing messages based on enterprise infrastructure requirements. Transformation may be required when domain data is stored in a very specific format that is different from enterprise-wide conventions. For example, domain-specific binary data may need to be transformed into an XML per enterprise policies.
 - *Firewalls* allow for enforcement of the domain encapsulation. Although, they do not perform any transformation, firewalls limit user access to only predefined points in the domain boundaries. Also, introduction of a firewall allows detection of services that do not follow encapsulation policy of a domain.

Figure 6 shows the infrastructure that may result from following the above steps:-

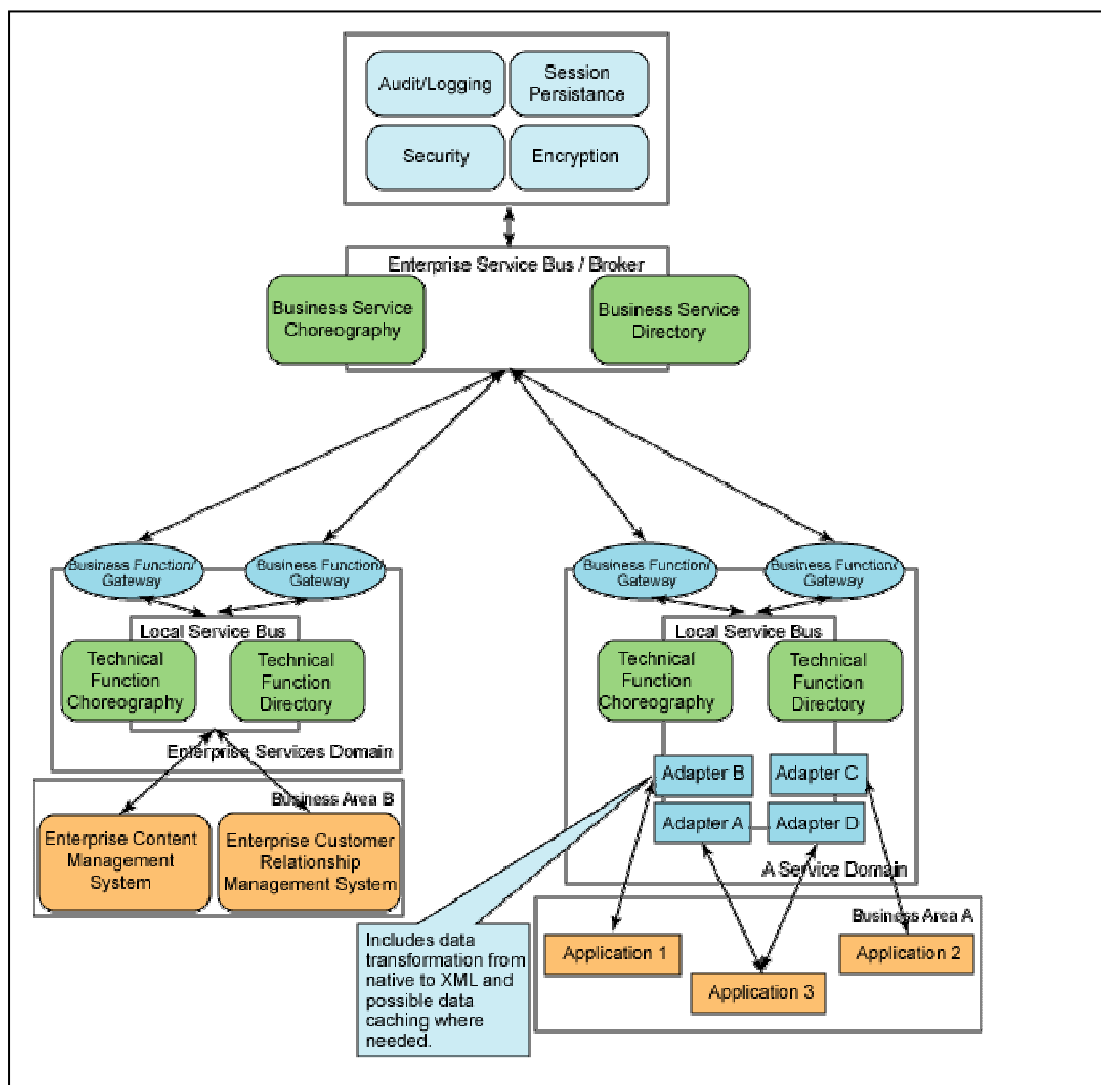


Figure 6: Defined and separated IT domains

Once all domains are bundled and business functions are defined, it becomes easier to aggregate them through the Enterprise Service Bus/Broker (ESB) and to employ Business Choreography to create higher level Business Processes and Transactions. Once the ESB is in place and the enterprise is functioning, the migration of legacy applications within domains may be performed with less impact on the rest of the enterprise.

Reduce migration risk

The phased approach to the SOA allows for easier mitigation of risk associated with changes to an enterprise IT infrastructure. Often, a company realises that supporting an old infrastructure is too expensive; however, drastic changes are risky. Splitting the enterprise into domains and introducing the concept of Local Service Bus (LSB) mitigates the risk and allows migration to a newer infrastructure in well controlled phases.

What all these mean?

It may not come as a surprise to many CEOs and CIOs that technology, regardless how advanced it may be, does not solve business problems despite marketing hypes from technology suppliers. The push for the SOA by technology suppliers to solve all kinds of business problem is nothing new – technology suppliers are known for their over optimistic claims. The reality is that SOA is no panacea. So, the key question is: how should CEOs and their CIOs approach the SOA? First, the company needs to make fundamental change to be service-centric. From our research, we found that there are more companies where the customer-as-adversary than companies where customer-as-advocate side. If companies win when their customers lose, then those companies are leaving the door open for someone else to come in and champion the customer. Despite claims by many companies that they treat their customers “as king”, our research of large to medium size companies over the last five years indicate that this claim is just not sustainable.

Once the company has made the fundamental transformation to be service-centric, then planning to adopt SOA makes commercial sense. Adoption of SOA is not an IT initiative that can be left with the CIO. Instead, the SOA initiative must be led by the business – for example, the COO and supported by the business unit heads together with the CIO and CTO. The success of the SOA is dependent on the leadership of the business unit heads and how they managed to create a SOA that serves the current purpose and is scalable to support the future business needs.

The technology industry is relatively young and SOA is the industrialisation of software development process. As every historian knows, revolutions rarely run as planned and implementation of SOA is unlikely to be different and there are plenty of challenges to be overcome. We saw industrialisation of many mature industries, such as manufacturing, lead to massive productivity gains but also threw out companies with impoverished craftsmen whose skills were no longer needed. The emergence of SOA is likely to produce similar outcomes: cheaper and more flexible IT systems combined with a painful adjustment for artisan-software developers and poorly performing software companies who get left behind.

NOTES

1. Mohanbir S. Sawhney, "Don't Homogenize, Synchronize". Harvard Business Review, July/August 2001.
2. "Surge in complaints over banks by public". Jane Croft, The Financial Times, May 1 2006.
3. See more details on SAFE™ and how to deploy it at Centrix website (www.centrixconsulting.com).

About the authors

Jon Fuller is the Operations Director of Centrix; Sukhendu Pal is a consultant.

About Centrix

Centrix is a leading independent consultancy that brings together the best of business, service and technology to create lasting value for its clients.

What Centrix Brings

Centrix designs effective operating formulae built on both innovative technical solutions and the practical realities of running a successful business. We help companies grow their businesses by working with them to: create services aggregation strategies that meet the requirements of stringent regulatory and compliance directives; design secure strategic offshoring and outsourcing platforms; develop service oriented architecture; reconfigure end-to-end processes, systems, and subscription based services to deliver greater operational efficiency. Centrix's approach is set apart by capabilities that help companies tie operating and technology decisions to what customers' value most. When we help our clients align their business to what really matters to customers, they achieve performance breakthroughs.