ABSTRACT

This paper illustrates a framework for Information Technology (IT) service development that translates the Voice of the Customer (VOC) into Service-Oriented Architecture (SOA) design requirements for the retail industry. Contributions of the work include the ability to evaluate SOA based applications in terms of the business value of the services they provide as well as the development of a framework that incorporates Quality Function Deployment (QFD) to the development, and improvement of SOA applications.

KEYWORDS: Service-oriented architecture (SOA), Quality Function Deployment (QFD), Voice of the Customer, Retail Industry

INTRODUCTION

Service oriented architecture (SOA) appeared on the business scene several years ago with promises of transforming business processes, as well as the field of information technology (IT). Using SOA, IT functions are decomposed into a set of standardized services that have well-defined basic offerings and standardized interfaces to allow flexible composition and reusability [8].

A recent survey reported that more than half of large enterprises had spent an average of $1 million dollars on SOA efforts, and at least 40 percent expected to spend $1 million more on SOA over the next 12 months. However, even though companies typically reported success in streamlining the IT function through reductions in the number of applications maintained and slashed IT budgets, they also reported that a disappointing 30% or less of SOA applications will eventually be reused or shared across business units. Furthermore, only 5% of companies expected to see more than half of their services reused [3].

Business executives have argued that SOA applications, like many previous information systems, lack clear connections to business objectives or detailed understanding of the nature and...
scope of business processes. In addition, meeting software reuse goals to any significant extent requires a structured design and planning process. In this sense, research is needed to address the recurring disconnect between IT and business functions that continues to exist, despite the potential of SOA based applications.

For that purpose, we propose the use of a structured process called QFD to assist in translating customer needs to technical requirements at every stage of the SOA design process.

**QUALITY FUNCTION DEPLOYMENT**

Quality function deployment (QFD) is “an overall concept that provides a means of translating customer requirements into the appropriate technical requirements for each stage of product development and production” [5]. At its core, QFD represents a hierarchical design process for product and process design, quality management and strategic planning. QFD deploys requirements through several levels of design and operation using a series of relationship matrices or houses.

Compared to traditional design approaches, QFD requires designers to spend more time defining the new system requirements and examining the ramifications of those decisions [4]. At the same time, QFD reduces time spent later in the process to revise the design, and promotes a better understanding of both the customer demands and the design interactions.

A number of applications of QFD to the field of information technology can be found in the literature. For example, Tan et al. [6] presented a simple application of QFD to IT systems, demonstrating the suitability of QFD in the design and development of IT projects. Han et al [1] developed a normative method that uses a QFD approach to evaluate the alignment between business strategy and information systems, though the authors did not address issues such as reusability. Overall, even though different authors have validated the use of QFD for the design of IT systems, there are no reported applications of QFD to the development, evaluation and improvement of SOA applications.

**PROPOSED FRAMEWORK**

The proposed framework uses a series of QFD matrices to translate customer needs into IT requirements, and then IT requirements into system modules. The required system components are then compared to the existing systems. From that point, it can be determined:

a) the gaps in existing systems to our existing products/services; and
b) the gaps in existing products/systems to systems that need to be created.

This is the point where traditional design methodologies would normally stop, but with the use of SOA principles we are interested in determining if what needs to be created can be used in other projects. For that purpose, we develop a commonality matrix of system modules. Figure 1 summarizes our proposed framework.
The underlying idea is that the use of common modules for different systems can result in lower costs for development and maintenance. At the same time, the use of common modules can also enable faster development of new systems. However, it should be noted that commonality can only be achieved when modules perform useful functions for a range of considered system types, and when the interface to the module is compatible across many different systems.

Without a structured approach to collect, organize and display the information about the modules that have been developed (including ways of usage, etc.), an application would lose most of the benefits that an SOA effort typically brings.

The goal of the final step of our proposed methodology is to develop a commonality matrix that describes ways in which different components are similar, thus supporting reusability. Figure 2 presents a Commonality Matrix that relates different customer needs to system modules already developed for past projects.

### FIGURE 1
Proposed Framework

### Commonality Matrix

The underlying idea is that the use of common modules for different systems can result in lower costs for development and maintenance. At the same time, the use of common modules can also enable faster development of new systems. However, it should be noted that commonality can only be achieved when modules perform useful functions for a range of considered system types, and when the interface to the module is compatible across many different systems.

Without a structured approach to collect, organize and display the information about the modules that have been developed (including ways of usage, etc.), an application would lose most of the benefits that an SOA effort typically brings.

The goal of the final step of our proposed methodology is to develop a commonality matrix that describes ways in which different components are similar, thus supporting reusability. Figure 2 presents a Commonality Matrix that relates different customer needs to system modules already developed for past projects.

### FIGURE 2
Commonality Matrix by Customer Needs

<table>
<thead>
<tr>
<th>Customer Needs</th>
<th>System 1</th>
<th>...</th>
<th>System i</th>
</tr>
</thead>
<tbody>
<tr>
<td>CN 1</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>CN j</td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
RETAIL QFD EXAMPLE

Service quality researchers have noted that a product’s design will be unsuccessful if the VOC, the processes, and the systems have not been cohesively incorporated into all aspects of service development [2]. In the retail industry this is of the utmost importance since managers have an enormous amount of information and data coming from multiple sources that have to be very well managed [7]. Retailers also face the challenge of the varying and fast changing dynamics of the retail industry. By using a QFD design we can graphically show how a business service provider can easily gather customer needs, transform these needs into IT requirements, compare these requirements to existing IT systems, and then determine the appropriate course of action with a modular design of the product or service.

Applying QFD to services is fundamentally different that QFD for products. This is due to the nature of services where the customer is seen as a coproducer and the intangibility of the services [2]. Since, the customer is part of the service production and consumption; it is difficult to separate the parts from the processes [2]. Laurette et al. identify weakness of applying classic production QFD approaches to services: 1) service quality is usually an emotional response and not tangible like a defective product and 2) that applications of service QFDs only consider the needs of the customers and don’t do beyond the house of quality to address processes, systems, etc. In this paper when go beyond the house of quality to look at not only the customers needs, but to also address the service functions and designs needed.

This example illustrates a service QFD approach to outlining information technology improvements/needs of a retailer. The areas of improvements have been identified and then translated into appropriate IT requirements. We use the House of Quality to display the cohesion of the VOC and the IT requirements. The customer in this example is not an individual consumer of a product; however it is a retail chain enlisting a business services solution provider.

Figure 3 depicts common requirements of a retail chain seeking to improve its store and labor management systems. We show, in this matrix, that there are common IT solutions that can be developed from differing customer needs.
In figure 4, a new design matrix has been created from the IT requirements and then compared to the existing legacy systems of the retailer to identify functional gaps. With this information, the service provider can easily determine if existing products or services are sufficient to handle the requirements or if they will have to develop a custom solution. In the case where the service provider is unable to meet the needs of the customer with existing products or services, they will be able to determine if a new solution can be developed modularly and with reusable system components.
Reuse is one of the fundamental aspects of SOA and by using a commonality matrix, introduced in the previous section, applications can be developed based on its business value.

CONCLUSIONS AND FUTURE WORK

In this paper we have developed a framework for using QFD matrices to build SOA modular components for the retail industry. We have shown how the Voice of the Customer can be translated into IT requirements and how SOA applications can be developed to represent true business value. By developing reusable system components, service providers can deliver solutions more quickly and of higher quality than before. Although SOA is still in its infancy, it is changing the way businesses provide services.

A natural extension of this research would be to develop a model to determine the optimal number of modules and the amount of granularity of those modules that would ensure reusability. The model could also be expanded by taking into consideration information on service levels and probabilities of changing the system.

REFERENCES


