



# Four Quadrant Analysis

---

Prepared By: Cindy Gregoire, TxMQ Practice Manager, Middleware & Application Integration Services



TABLE OF CONTENTS

**Using Service Oriented Architecture** ..... 3

**SOA Requirements** ..... 3

**Leveraging the Flexibility of SOA**..... 4

**The Y2X Flowdown**.....5

**The Role of Business Analysts** ..... 7

**Four Quadrant Analysis: Business Analysts** .....7



- Are you having difficulty getting your SOA off the ground?
- Are business initiatives dragging down your infrastructure with lots of low-quality web services that you wouldn't even consider for reuse?
- Are you able to realize your SOA investments with rapid development and high business value and high acclaim from your business counterparts?

## USING SERVICE ORIENTED ARCHITECTURE

*It may be time to consider changing your requirements gathering process.*

There is a principle in developing solutions that bring business value for multiple business units called four quadrant analysis. This analysis involves the interview and collection of information gathering into four distinct categories or quadrants that help to bring together a more complete architecture framework for completing business process management initiatives leveraging your middleware services and application provisioning within the framework of a service oriented architecture.

Within a service oriented architecture (SOA), the style of developing and deploying applications involves an assembly of reusable components which are modified for a new purpose – the goal of which is to minimize development efforts, thereby minimizing the extensive needs for testing, and therefore resulting in rapid delivery of business solutions. Delivery of business automation involves the implementation of software to perform work upon business data. In most cases, it also involves the routing of decisions, inputs, or outputs to various user groups who operate independently from one another or may involve service providers or external business entities.

## SOA REQUIREMENTS

In implementing a SOA, requirements may originate from a number of sources making the job of the business analysis even more difficult as analysts attempt to identify and define what is needed to realize the SOA investments for flexibility, reuse, and speed-to-market.

Functional requirements obtained through agile development efforts tend to have a myopic focus on the user interfaces as requirements become known during an iterative process between application users and “agile” developers leaving BAs drowning in a mire of minute detail around the many options of where, how, and when to display fields, fonts, and typestyle themes.



Requirements that are defined through the Agile Methodology tend to be end-user focused with “look and feel” playing the higher priority than efficiency of code, system performance, or keystroke/user efficiency behaviors. Such requirements are labeled “non-functional” and generally become background information as projects get closed out based only on meeting requirements created based on the application web graphical user interfaces or originating on one of the many portal technologies.

Portal technologies are patterned for language and usage patterns which can be modified quickly providing a unique look and feel for user groups. Department portals may be modified without development code and optimized for the user groups accessing them, however, storage of business data and critical information can cause problems across the enterprise as decisions are made on an application-by-application basis (as they often were in pre-SOA days).

The issue comes in as data, process, and procedures now vary considerably by business process or department with information needing to be supplemented, indexed, or compared against metrics for proper handling or escalation surfacing all over the organization in portals, applications, Excel spreadsheets, Access databases, etc., whereby analysis data is not maintained or shared beyond an individual or department level source – and may not even be known by others outside the department or business function.

As a result, simple changes to the organizational structure can result in major loss of critical data (wiped off an end user’s desktop), immediate retirement of business assets (through non-use), or the requirement for entire groups to “re-tool” (subjecting the company to further inefficiencies and delays) to require use of only certain applications or interfaces used by priority groups.

## **LEVERAGING THE FLEXIBILITY OF SOA**

In an environment such as the above, how do you analyze the functional and non-functional requirements required to effectively leverage the flexibility and capabilities of the emerging SOA technologies without causing continual chaos in your service delivery?

How do you recognize applications which should be designed as common services across the enterprise? How do you manage proliferation of applications that all handle similar data – duplicate data – and end up requiring you to host large farms of application servers hosting undocumented applications with unknown owners handling data when you are not sure what the applications are, who is using them, or what they are being used for? These are common consequences of introducing SOA technologies without 4Quadrant Analysis and they can result in expensive and extensive rewrites.



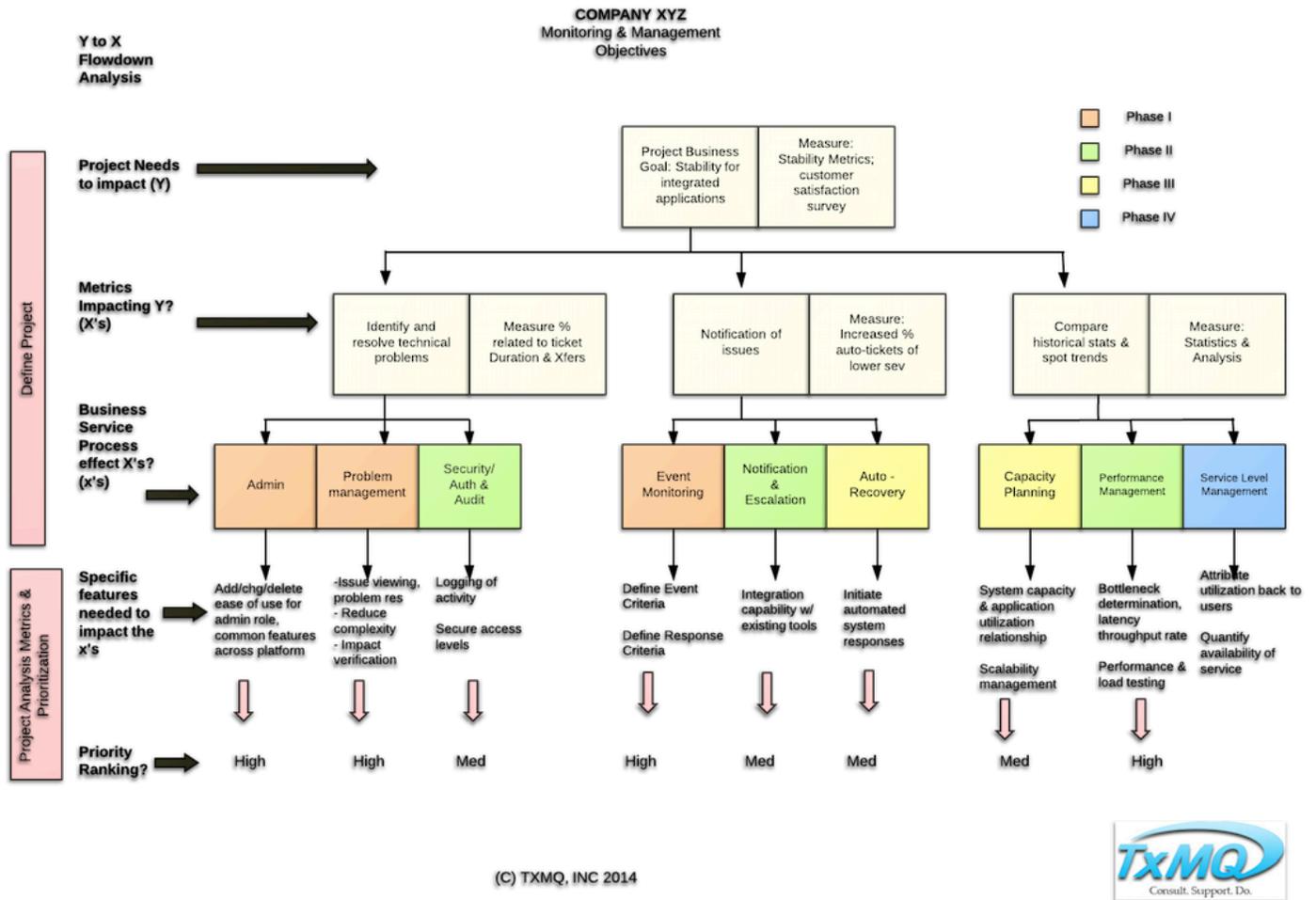
## THE Y2X FLOWDOWN

Enter: Four Quadrant Analysis – a new perspective on information gathering and requirements modeling that addresses issues introduced by the SOA space leveraging Six Sigma methodologies. This technique involves first and foremost, interviewing key stakeholders of your SOA initiatives for first and foremost, prioritizing the key objectives for the company’s SOA framework using the Y2X FlowDown tool.

The Y2X FlowDown organizes the project objectives and measurements with outcome in terms of quality metrics, service deliverables, dependencies, and creates a visual diagram from stakeholder input for the identification of project stages with measurable objectives that will be realized over the completion of one or more projects. During the Y2X FlowDown meeting, it may become apparent that some of the project objectives will not be realized during initial phases of a SOA. This is an important step during project initiation to ensure expectations are being managed realistically and in the appropriate context of cost management and dependencies with infrastructure projects that lay the foundation of service oriented architecture. The Y2X Flowdown also ensures quality metrics are discussed and identified at the earliest point of conception so that the project deliverables can be modified accordingly.

As a result of the Y2X FlowDown analysis, it may be found that additional software, hardware capacity, or major infrastructure investments may be required to “measure” the successes of the project. Additionally, it may become clear through the flow down discussion that not all needed stakeholders have been identified or involved in the discussions. The value of the Y2X FlowDown process is twofold: (1) getting stakeholders on the same page with project outcome expectations, and (2) the Y2X FlowDown diagram, an example of which is noted below in figure 1 – which will become a reference point referred to again and again during each project, and at the beginning and end of project phase as a “roadmap” of expectations involving both functional and non-functional high level requirements for the project.





Initially, the Y2X Flowdown seems like an extra step, however, used consistently across SOA projects, patterns begin to emerge with the flowdown process uniting SOA sponsors with affected infrastructure and development teams in an outcome-focused planning effort. It simply answers the question, what will be delivered, how will success be measured, and when can I expect delivery (project phase)?

Once the Y2X Flowdown diagram has been created for each initiative, the specific objectives and how they will be measured become a regular discussion point at all stage gates using the diagram as a key project management tool between the various groups that will be impacted by the new SOA, development approach, and process for rolling out new business capability on your SOA.

From this point forward, your service delivery should be governed by a process which prioritizes and leverages the SOA components and patterns which are performing well within the SOA. This is key to manageability and obtaining the promises of SOA.



## THE ROLE OF BUSINESS ANALYSTS

While business initiatives and infrastructure projects are being identified and prioritized, SOA business analysts are continually re-assessing critical business processes for process improvement opportunities. If they are not working on continual process improvement, there may instead be focus on the creation of much duplicity within the capability, thereby canceling out the return on investment of the SOA technologies. For example, defining services too granular results in no reuse. Defining services too coarse grained may result in slow processing and overburdening of the system with extraneous data.

Service definition training is critical for BAs to understand the trade-offs in a SOA. Many BAs design services like applications with data mapping requirements that are too myopic, or able to leverage other existing services to formulate fast-performing composite services. In many companies, the BA function focuses on end-user requirements while ignoring critical details such as throughput, error handling, escalation and routing of exceptions where such frameworks become the basis for Business Process Management optimization.

The role of the business analyst is to quantify the “as is,” the “to be,” and to prioritize the value of filling the “gaps.” Outputs may consist of activity diagrams, SIPOC diagrams, requirements models, and data flow diagrams. Many companies use “swim lane” diagrams to portray the business process, however, these diagrams and information become difficult to review, and can even become a source of confusion when a change in business flow occurs, or a new packaged application is purchased, especially if they contain data at too granular a level.

If start and stop points, inputs, and outputs of each business process are not easily identified, process improvements are impossible to identify and the business process is then mapped to how the “new” application works, creating yet another source of information overload for business users when the next business service is rolled out. This common business approach is responsible for creating even more sources of duplicate data across the enterprise, with multiple groups performing similar tasks (duplicity), and can lead to major out-of-synch data and data management nightmares.

It is at this juncture that many companies begin to take SOA governance seriously as a business service development approach. It should be considered that such confusion resolution must belong to the BA role. Problems that are created because of duplicate data, duplicate functionality within applications, and business functions being performed in multiple places in the organization – those are all problems that require solving by a BA.

### FOUR QUADRANT ANALYSIS: BUSINESS ANALYSIS

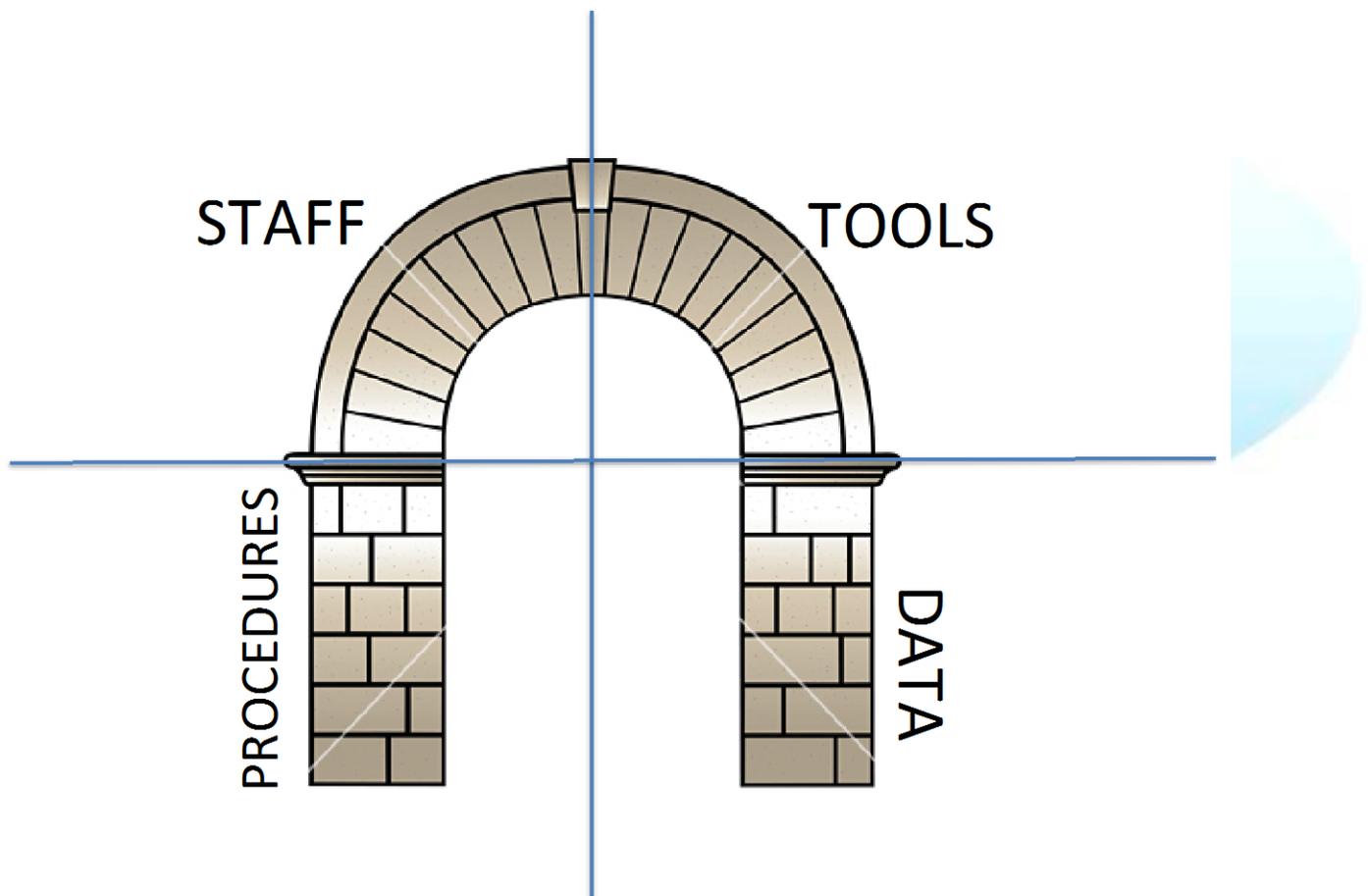
The tool of the Technical Business Analyst (TBA) is the collective assessment and evaluation of technical requirements into four quadrants. The TBA is a role which is commonly being filled by what is more recently known as the Enterprise Architect as enterprise level tools, data, and services are being defined



for the sake of leveraging a common toolkit across the enterprise. Such efforts result in introducing ERP technologies such as SAP, or outsourced functions such as payroll and accounting. The TBA is able to approach the organization holistically by building a visual that helps the organization to understand its use of technology with the understanding that businesses run effectively because of people following procedures, accessing tools which use corporate data – whether such staff are internal, partners, or outsourced.

These are the four quadrants in this model: People, Procedures, Tools, Data. There is another element in the model to bridge where there are groups that use the same data – those points are “bridged” by what we call Architecture, which visually speaking can take the form of figure 2.

Figure 2:



For each department in the company, these four elements are assessed by the TBA – starting with the critical business process activity diagrams. The TBA becomes quickly knowledgeable about which department requires what information, using which tools (applications and services), to complete which procedures (business processing). This approach prevents “compartmentalization,” whereby the focus is



quickly lent over to a specific need (where the squeaky wheel gets the grease), rather than its relevance to particular business functions which all have quantifiable value to the overall business process.

This fundamental visual of the organization provides the “connectors” to be quickly identified between departments, between roles within the organization and escalations within a business process, critical dependencies, need for SLA and time-critical processing, and need for data maintenance standards to control and manage enterprise data. The “bridge” between staff and tools is the “keystone” or what we commonly refer to in IT as “architecture.” This distinct approach leads to the detection of existing assets and modifying them for multi-use functions, preserving integrity of data, decreasing the need for data maintenance, and keeping controls in place that regulate efficiencies. When incorporated into business requirements analysis, there are few “dropped balls” or missing requirements when all four quadrants are addressed methodically.

Where connectors are identified, the need for architecture to quickly bridge between quadrant elements, mapping of capability to applications and services, while quickly providing the most effective use of your cornerstone technologies for an organization are key inputs to the enterprise architecture which enables rapid service deployment across business units.

This approach can also speed the creation of the critical inventory of enterprise assets that can be reviewed for optimizing vendor relationships, consolidation, and allow you to realize exponential cost savings as you streamline your assets, redesign and realize your greatest business optimization ever!

