
Exploring Services and Requirements – Part one

Author: Charles Edwards

Version: 0.09

Date: 10th July 2008

Abstract

As the architectural concept of Service Oriented Architecture (SOA) is implemented in many organisations, it is starting to introduce somewhat of a dilemma between the Business Analysts and Architects. People in these roles are wondering just how the Business Services, Business Processes and Use Cases all integrate and fit together.

Most systems are implemented as projects, with the scope limited to the associated requirements of the project. However these days, in order to maximise Service re-use, the business community in the enterprise have to think about and define their Business Capabilities and Requirements at more of an Enterprise Level rather than just at a particular System or Project level.

The aim of this series of papers is to explore the different semantics and terminology to align the areas of overlap between the various concepts, in order to suggest an overall simple integrated approach that works satisfying both the Business Analysis viewpoints and the Enterprise Architecture viewpoints.

Introduction

For as long as we have been building software systems, implementing them in the business has been very project centric. The business requirements have been defined in order to satisfy a particular project at a point in time. Often projects have not considered the bigger picture and have even been in isolation of all other factors in the enterprise. For example, cases where two projects have each specified very similar business requirements that get implemented in different systems by different projects, causing resource wastage, process duplication and inconsistency. This leads to a higher maintenance burden and costs more time and money.

Service Oriented Architecture (SOA) implementations in organisations have started the process of centralising the Architecture of each of these project solutions into one cohesive solution stack. This mechanism helps converge the architecture and simplify the complexity, time, cost and agility of going to market. So the idea is that as the business changes, any new business architecture requirements that arise can do just that, extend or re-use what already exists. It sounds simple, but we have a few problems and challenges to overcome.

The Problems

These problems are a hangover from the approaches taken to date, which prohibits an enterprise wide approach:

- **The way in which Projects are economically justified** – we do not look at long term or wider benefits.
- **No Business Architecture** - Does a well described Business Architecture ^[1] with defined Business Services exist in many organisations? Probably not. OK - very few.
- **No Repository of business operational knowledge** - Once projects finish, do some of the coarse grained requirements, business analysis outcomes, structures, models and contexts ever make it back into a central Business repository to baseline the business architecture? Probably not.
- **Map requirements to Business Architecture** - Do the BA's and other requirements gatherers know how to map their requirement specifications (Use Cases, Requirements, etc.) to the existing Business Architecture context? And specifically to the Business Services?

¹ **Business Architecture** is loosely defined as an Enterprise wide, context (high level) model, of the behaviour and structure of the Enterprise.

Some of the reasons for this state of affairs are:

- **Short term view** - Costs are perceived easier to allocate to a particular project, than to an Enterprise wide fixed overhead that would bring longer term business benefits and value.
- **Project Isolation** - Requirements are **Project centric** and don't see the bigger enterprise wide picture.
- **Cost and Value justification** – The business still find difficulty in cost justifying a model of the Business Architecture. They battle to see the value. Some have tried and failed, which hasn't helped.
- **No Enterprise Architecture** – For the above reasons and due to the immaturity of EA in general the Enterprise Architecture functions tend to be rather IT centric or understaffed, even non-existent. This fortunately now appears to be starting to change for the better.

Enterprise Architectural Services

Before starting, let's describe what we mean by the Architecture of Services for the Enterprise, managed by Enterprise Architects (EA's); *Services* are an architectural layering of enterprise services that represent both unique and re-useable components, as follows:

- The **architecture of the business** as a set of:
 - **Business Services.** This is part of the structure (static) and behaviour (dynamic) specification of the business. Think of business services as part of the context for any project. E.g. *Place order service*. This service has structure, behaviour (business processes) and can be both a business requirement and/or a component of the existing business architecture, with invocation, value, etc.
- The **architecture of the solution systems** as a set of
 - **Information Systems Services.** These services define the structure and dynamic behaviour of the automated information system solutions implemented for the business. i.e. Software. Some example information systems services could be *Place_order()* and *Confirm_receipt()* that are part of the more coarse grained service *Customer_Services()*. *Place_order()* may use some other info system services such as; e.g. *Authorize_payment()* being a part of the *Payment_services()*.
 - **Infrastructure Technology Services.** All of the above are supported by a set of infrastructure technology services some examples of which might be 'a *Database Service*', 'an *Application server service*', on 'a *Virtualised hardware server service*', using 'a *unix operating system service*', 'a *WAN network connectivity service*' and 'a *Storage area network service*'.

Three views into the same thing within the Enterprise

Let us zoom into three view points of the enterprise and look at each from their own perspective.

One type of view per page with an associated diagram to make easier reading.

1. The Project 'Business Analyst' Viewpoint

In one part of the office, the Business community, work with Business Analysts and Requirements gatherers to specify project requirements. This is project specific, and in all likelihood happens in isolation of other projects.

Typically requirements capture does not happen in a formal context of an existing well defined Central Business Architectural Model, following a Business Services Taxonomy and/or enterprise wide Business Process or Business Use Case Model. (Definitions of these terms to follow in subsequent paper.) This seems to depend on the industry, the organisation's maturity; but certainly very weak in the large financial institutions I've worked at anyway. Requirements capture happens on projects and tends to start from scratch and also often dives down into the minute details very quickly. Without a persistent central business model explaining the business structure, processes or context for the project to use, the enterprise experiences virtually zero benefit of any reuse or formal context setting. The context of each Project is usually embedded in the minds of people on the project, from meetings, emails, discussion. The context of the business knowledge is largely in the heads of the business community. Both of these contexts are seldom easy to discover for someone new, someone late into the project or outside the project.

In the figure below, you will see the main role of the Business Analyst shown in red within the context of the enterprise. This role spans from the business community, through actual business processes they do, into the IT community. They formulate detailed requirements for the Software Development (or COTS acquisition) project. They are flanked by a Project Manager and a Solution Architect. They concentrate on a *narrow project scope and depth of detail or fine grained granularity* to help ensure the project eventually gets into operation to run live with the business requirements satisfied. Shouldn't the Product Owner do this? (See challenges section below)^[reference3]

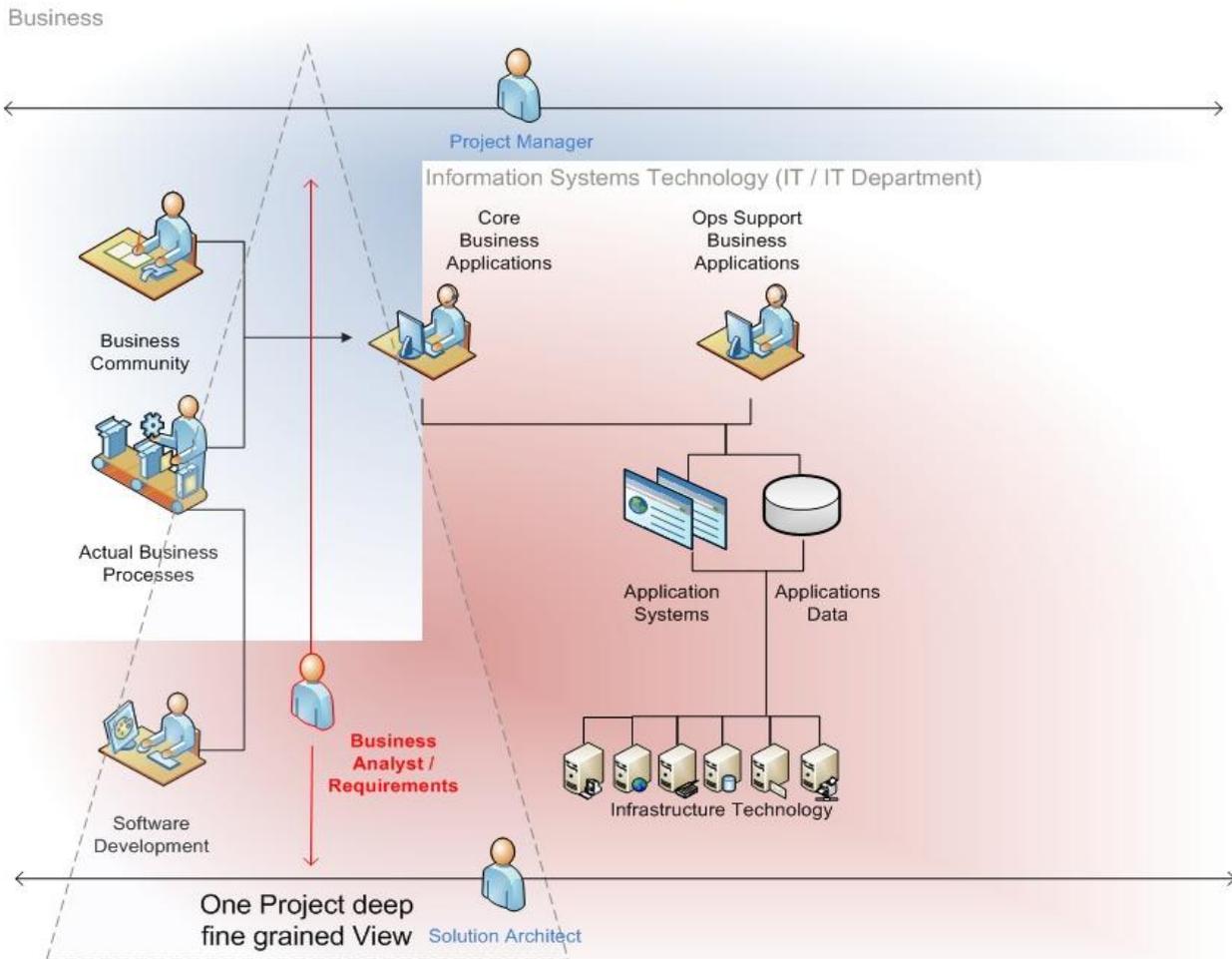


Figure 1 - The Business Analyst Viewpoint (typically on one Project)

2. The Enterprise wide 'Business Architects' Viewpoint

In another part of the office (think: duplication, isolation, lack of proper communication), the Enterprise Business Architects define Business Architectures with all the relevant layers and types of Business Services, Business Use Cases and Business Object models, with the ultimate aim of defining a portfolio or catalogue of business services that can be defined, re-used and governed within the enterprise over time.

If you look at the figure below, you will see the Business Architect highlighted in red as the main role in defining the Business Services. This role sits amongst the other roles in Enterprise Architecture also building up the Enterprise Services. The other main roles are shown in bold blue; notably the Service Architect and the various Solution Architects from various projects.

The Enterprise Business Architects aim is to get the *breadth of the Business Services defined but at a shallow level*, in order to define a structural framework model (Business Architecture) that defines the business. This can be used by anyone else, but mainly the Business Analysts and Solution Architects, to obtain the context of the existing business and systems as they change over time. This ties in closely with Business Strategy too. It's about alignment, not just being a passive body of knowledge. These architectural models are versioned as they change and regularly base-lined, so that the business changes are change controlled and configuration managed.

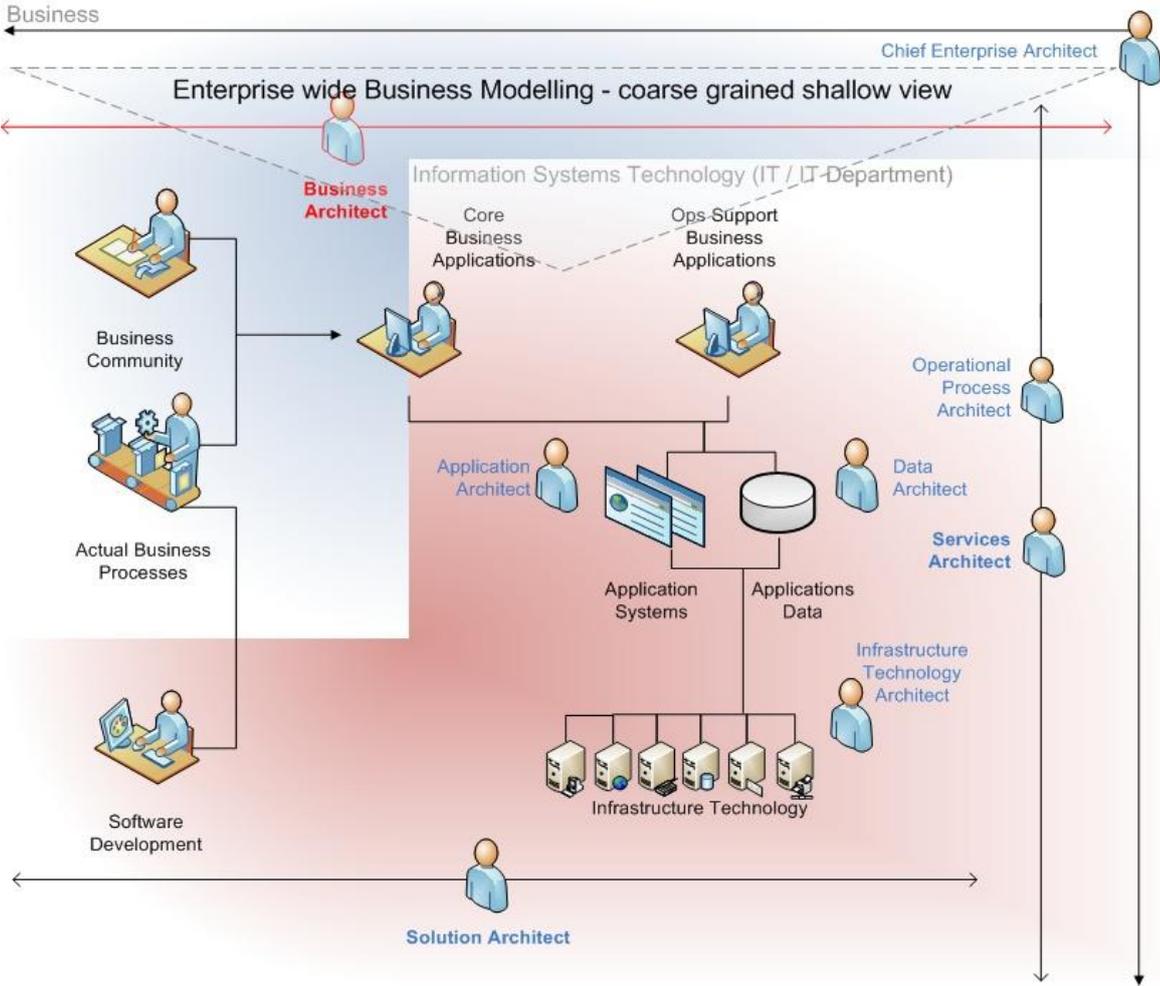


Figure 2 - Enterprise Business Architecture Viewpoint (Enterprise wide)

As opposed to the business analyst's viewpoints where there is a distinction between Problem and Solution, by the nature of Architecture work, there is no real boundary between defining the problem and specifying the solutions. The architects deal with what currently exists, what is strategised for in the future and any segments of solutions at points in time along the way. Enterprise Architect's viewpoint is more of a current state that has to morph over to a future state in a series of segmented projects, with inputs being the risks to the business and the requirements and changes the business need at those points in time.

3. The Enterprise wide ‘Service Architects’ Viewpoint

Independently, the Enterprise Service Architects (with Solution Architects from all projects participation), define their SOA architectures with all the relevant layers and types of Services, with the ultimate aim of defining a portfolio or catalogue of services that can be defined, re-used and governed within the enterprise over time.

If you look at the figure below, you will see the Enterprise Services Architect highlighted in red as the main role overseeing the definition of the SOA Services for all service layers Business services, Information systems services and Infrastructure Technology services. This role sits amongst the other roles in Enterprise Architecture also building up the Enterprise Architecture. The other main roles are shown; notably the Business Architect and the Solution Architects from various projects.

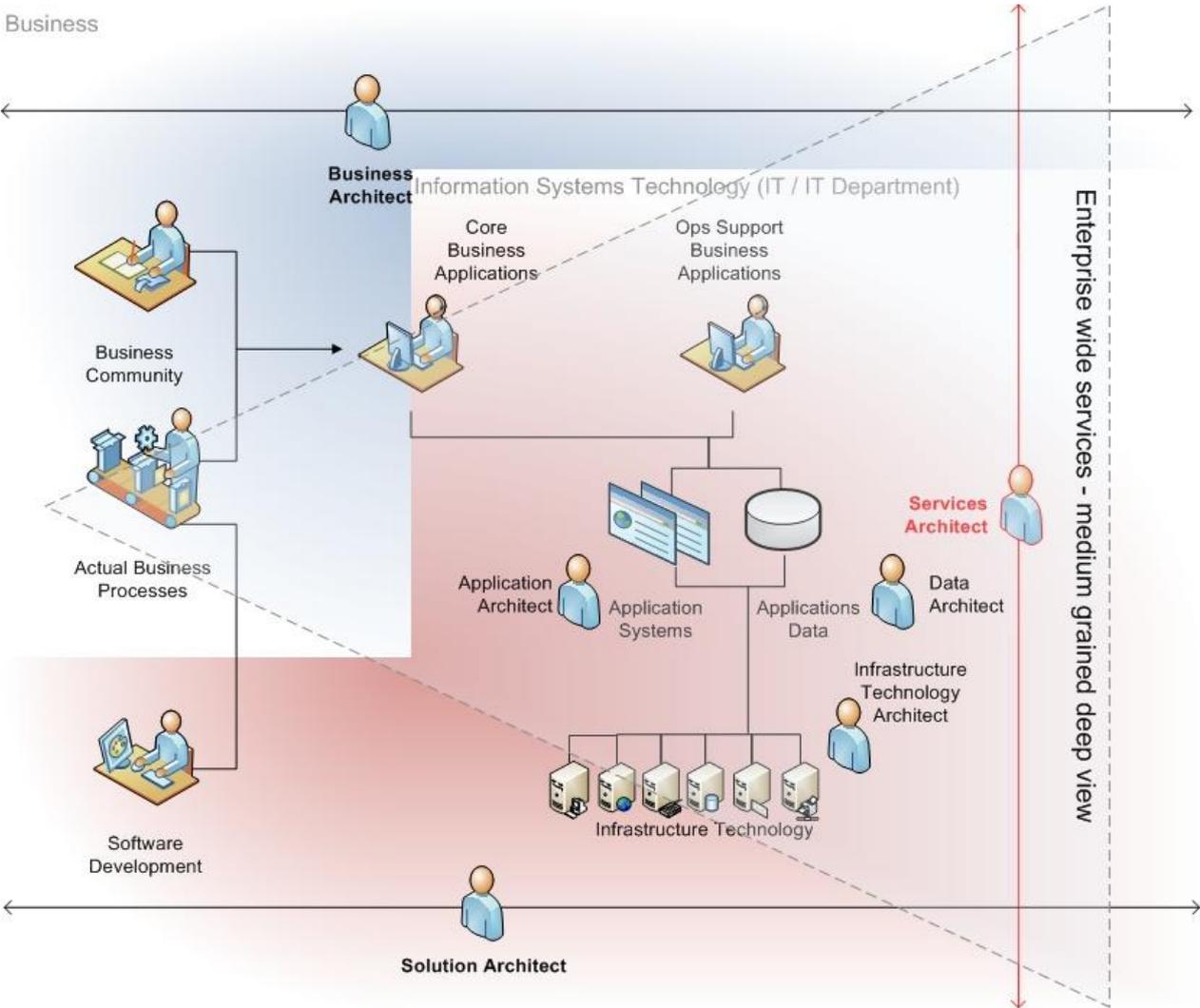


Figure 3 - Enterprise wide Services Architecture - medium grained and deep view

The Enterprise Service Architects aim is to get the *medium grained view of all Enterprise Services*, in order to define a structural framework model (Enterprise Services Architecture) that defines the business. This can be used by many, but mainly the Business Analysts and Solution Architects. The services also impact the Application Architects, Data Architects and Infrastructure Architects, who need the context of the existing business, systems and infrastructure as they all change over time. These architectural models are versioned as they change and regularly base-lined, so that the business changes are change controlled and configuration managed.

The Challenges

Varied approaches exist which all offer some value; some have their roots in the Business Process Re-engineering (BPR), some from Business Process Modelling (BPM), some are rooted in Enterprise Architecture (EA), some are newer in Service Oriented Architecture (SOA) and some rooted in Software Development and requirements management.

Adding spice to life, tensions still exists between:

- The traditional Enterprise Architects viewpoints versus the newer SOA Architects viewpoints that are not yet fully resolved and causing confusion. ^[footnote 2]
- The traditional Business Analysts versus the newer Agile software development community who feel the Business Analysts' get in the way and the Developers should be talking directly to the Product Owner business to obtain this detailed information to both speed up the process and make it more agile ^[footnote 3].

How do we optimise and align these three approaches to offer the best result for the business? How will the parties work optimally together? How do we get away with the least amount of work which offers the most value quickly? Something practical, useful and within the scope of a few people, so regularly updated and kept valid.

Meeting the Challenges

To meet these challenges the Business Architects and other Enterprise Architects have to work closely with the Business Analysts and Requirements gatherers.

Based on what has been described so far and looking at the diagram on the following page, the main convergence in approach between the roles is key:

- **Business Analysts**
 - **Values** = Solve the problem quickly. Rush. Now. Deliver. This project narrow view only. Cut out all else. Work around integrations. Live and breathe detail. Verbal Communication. In memory or emails. Do what it takes. Get it done. Go.
 - **Look for Depth, at fine granularity (narrow but deep)** - Business Analysts on Projects work within a specific scope (narrow but deep) capturing a fine granularity of detail and only for that Project.
 - **Do Project specifics and focus to do the job quickly** - Business Analysts on Projects work on projects, and focus on the detail of those projects, so that they can get the job done quickly, and avoid concerning themselves with non-project related external issues.
- **Enterprise Business Architects**
 - **Values** = Short, Medium and Long term thinking. Build up central context knowledge base of the Architecture of the organisation and behaviour for the organisation. High level. Current and Future. Broad and Wide. Traceable for impact analysis.
 - **Look for enterprise Breadth, at coarse granularity (wide but shallow)** - Enterprise Architecture needs to work within an overall broad context (wide but shallow) capturing a low granularity of detail but for the whole Enterprise, so that all Projects can align and converge to this business framework it defines.
 - **Supply 'Context' for business re-use, governance, convergence and refinement** - Enterprise Business Architecture gathers and provides a framework and context for all Projects. A Business Architectural Model at all levels is provided, so that everything traces and relates, so that change, dependencies, risk and impact can be managed. This helps converge, govern and refine the business processes and make the business a more efficient organisation over time.

^[footnote 2] See the blog [Reference 1] and presentation [Reference 2] below for a separate discussion on this particularly accurate observation.

^[footnote 3] See the excellent article [Reference 3] in the references below. I tend to agree with the observations, but hey don't shoot the messenger! I do wonder about the capturing of the more detailed information and knowledge into a central model though, although that probably becomes the Product Owners role.

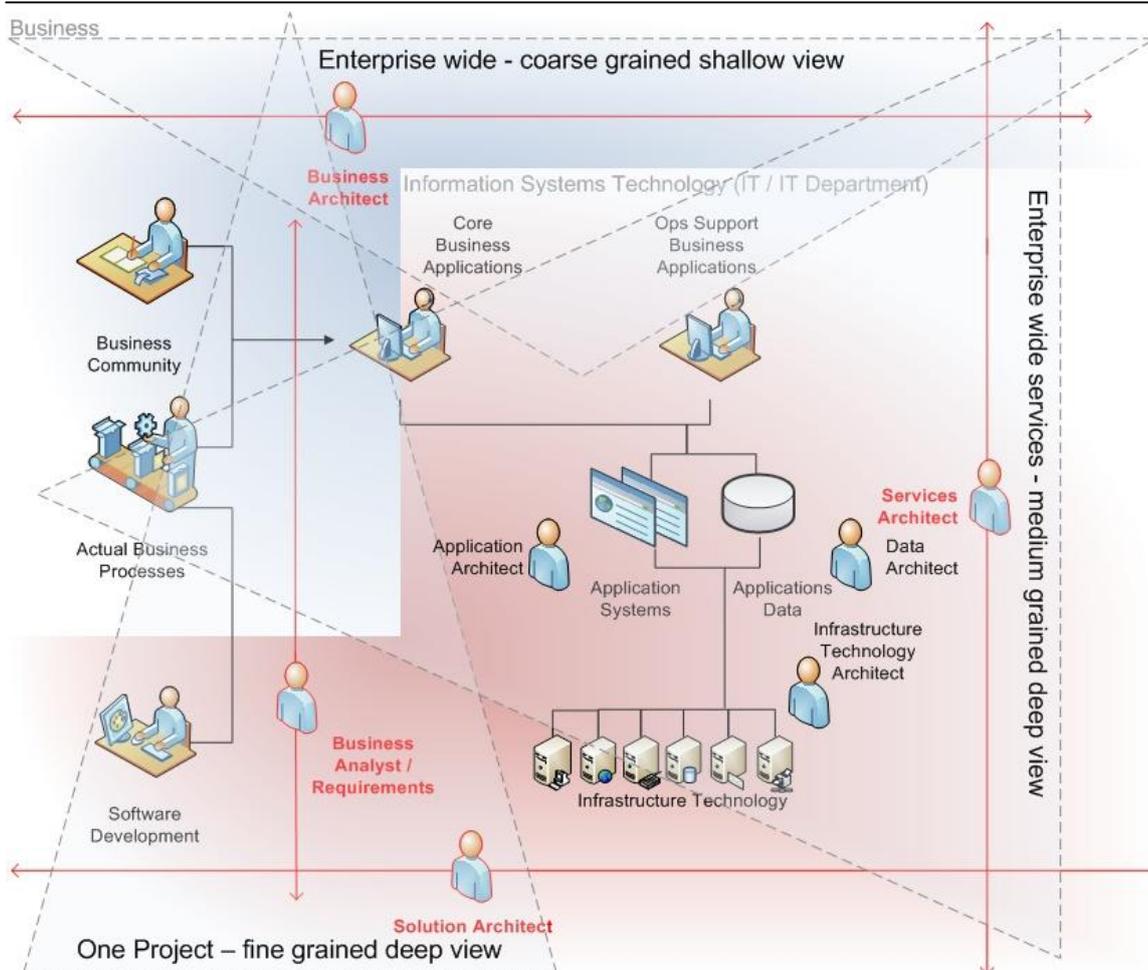


Figure 4 - Getting the best collaboration between all Architects and Business Analysts

- **Enterprise Service Architects**

- **Values** = Tie it all together. Traceability. Version control and configuration management. Make it work as a single cohesive system. Maximise future quick re-use. Make the Enterprise Agile.
- **Look for complete coverage, at medium granularity** – This role helps build a central model over time that threads business to information systems to infrastructure technology with version control.
- **Supply 'Context' for all re-use, governance, convergence and refinement** - Enterprise Services Architecture gathers and helps provide a framework and context for all Projects. An Enterprise Services Architectural Model is provided at all levels, so that everything traces and relates, so that change, dependencies, risk and impact can be managed. This helps converge, govern and refine the enterprise and make the enterprise a more efficient organisation over time.

Next time

This series of papers stops short of defining the Roles interactions as described above, but does go on to cover the concepts of what might be specified and how the concepts inter-relate.

Next time we will define the various concepts of 'Services' and remind ourselves about the concepts of Business Modelling and Requirement Specifications to see where the overlaps are so that we can explore at ways of merging the concepts and make them work together.

After that we also explore the similarities and overlaps between the concepts of Business Processes, Business Use Cases and Business Services.

References

- [1] 2007, October 15: Blog posted by Andy Mulholland in [Architecture](#)
http://www.capgemini.com/ctoblog/2007/10/enterprise_architects_versus_b_1.php
- [2] 2007 July 19: David Linthicum – When EA and SOA worlds collide!
<http://www.eadirections.com/uploads/file/When%20EA%20and%20SOA%20Worlds%20Collide.pdf>
- [3] 2007 March 3: Ambler, Scott: Rethinking the Role of Business Analysts: Towards Agile Business Analysts?
<http://www.agilemodeling.com/essays/businessAnalysts.htm>

Contact details

Charles.Edwards@processwave.com

www.processwave.com & www.AgileEA.com