

# A Simple Guide to Enterprise Architecture

Ian Bailey

This paper provides a summary of enterprise architecture, outlining what it is, what the benefits are, and a few pointers towards best practice. This paper also touches on the subject of enablers such as architecture frameworks and meta-models for those readers who want to delve deeper.

All content of this document is © 2006 Model Futures Ltd. unless otherwise stated. Any unauthorised copying or re-use of this material will be dealt with swiftly and mercilessly.

Model Futures is a consulting company specialising in Enterprise Architecture, Data Modelling and Ontology

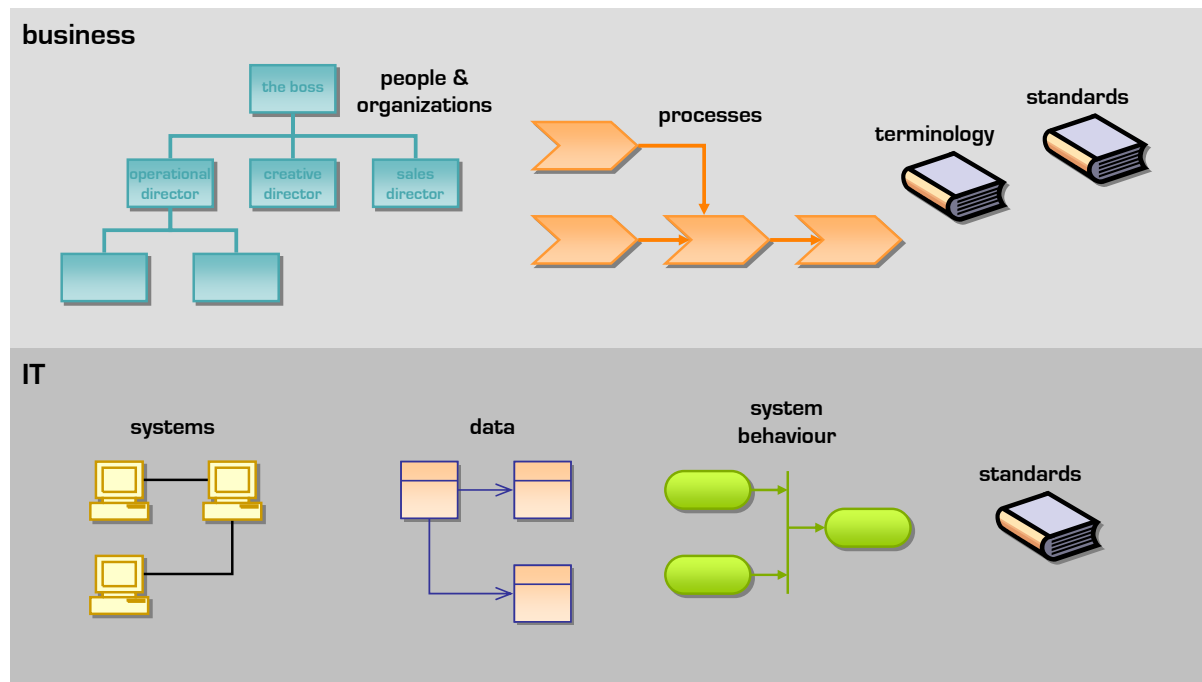
[www.modelfutures.com](http://www.modelfutures.com)

## Introduction

The practise of enterprise architecture (EA) is about informing the business decision making process by understanding how complex organizations are structured, how they function, and what technology supports those functions. It would be wrong to think of EA as an IT discipline, as it is far broader than that. However it must be recognised that, in most cases, increasing complexity in IT has been the driver for EA adoption – the sudden increase in IT portfolio size that results from a merger or acquisition has often been identified as a justification for EA projects.

What EA seeks to do is model the relationships between the business and technology in such a way that key dependencies are exposed from the underlying complexity and so can be used to support decisions. One could call this a *critical path of dependencies* between the operational domain and the technology. For example, by ascertaining which business processes are supported by a given software application, and then which platform the application runs on, it is possible to calculate the impact on the business of hardware obsolescence. From the same model, it would also be possible to examine, the cost of replacing the application, or which processes could be best re-designed to reduce the IT portfolio. When other dimensions are introduced such as data models, organizational structure and standards, questions which initially appear impossibly convoluted can be answered, provided the dependencies between domains are all known.

The information that is used to inform an EA can often be obtained from the appropriate departments of the organization (though the information can vary in quality). For example, the IT department will probably have a reasonable model of the application deployment throughout the organization, HR will probably maintain an organization structure, and the ubiquitous management consultants will have modelled the organization's processes at least a dozen times.



Clearly, the task of assembling the information for analysis of an enterprise of any significant scale is not going to be easy. For this reason, EA is never undertaken without first having a question (or preferably a set of questions) whose answer would provide sufficient business benefit to justify the cost of analysis. EA is *not* an audit exercise, it is a decision support process. And, once one has gone to the trouble of producing a model on an enterprise scale, it is usually more economic to maintain it than let it "go stale" and have to conduct the whole analysis again next time a question is asked.

As a final word of introduction, it is worth clarifying that "enterprise" usually does not mean "corporate-wide". The scale of the enterprise architecture is decided by the boundaries set by the analysis – it could be a department, a project, a team, or a government of a country.

## Why Produce an Enterprise Architecture Model ?

It is tempting to say that if the head of an organization doesn't see the need for an EA model, then it is probably a simple enough organization not to need one. However, the truth is that most enterprises started out small and simple, where one person could genuinely understand the whole thing, but grew to the point where no single person could describe the whole. The same can be said of the corresponding IT systems that support the enterprise. The problem is that the people who run these enterprises are loathe to admit (or often don't even realise) that the level of complexity has grown beyond their ability to comprehend. This culture of denial is behind many of the management disasters of the information age – assumptions based on poor understanding of the key dependencies, weaknesses and bottlenecks in an organization have resulted in poor decision making. In the last few years, there has been a spate of corporate mergers of very large companies. The merged organizations each had their own cultures, processes, terminology and technology. Understanding how best to rationalise these organizations is only possible using models at an enterprise level – hence the recent upsurge in interest in EA.

The benefits of EA are not always easy to measure – anyone with experience of change projects in large organizations can see that an EA approach is valuable, but it is hard to put a monetary figure on the benefits. Even just concentrating on tangible projects such as IT delivery, it is hard to put a figure on the benefits of EA. There are many opinions on why large IT projects tend to fail so spectacularly, but running through all of them are some key threads:

- Gaps in understanding – the business community are unable to elucidate their requirements to the technology providers, and the technology providers are unable to describe the impact of technology changes in such a way that the business community can understand the risk.
- Weakness of verbal requirements – written specifications are open to interpretation and contribute to gaps in understanding
- Lack of project cohesion – it is very difficult to coordinate a large development team to work towards a singular goal when their allotted tasks are specialised and isolated

It is clear that a well executed EA approach helps to overcome all these problems, but this still a “gut feeling” rather than anything that can be measured. The only way to measure the benefits is to compare two almost identical projects, one which used an EA approach and one which did not. Just such an analysis is documented in a paper by Tony Brown<sup>1</sup> where two IT implementation projects are compared. Each project implemented a worker's compensation system for US states (one of which was Ohio). The Ohio project came in at \$3.5m and was delivered in three years, whereas the other project cost \$42m and took 12 years. The projects were comparable in scale and used similar CASE technology – the key difference was that the Ohio project took an enterprise view and so was able to make smarter decisions about implementation because the delivery team better understood the requirements of the business.

---

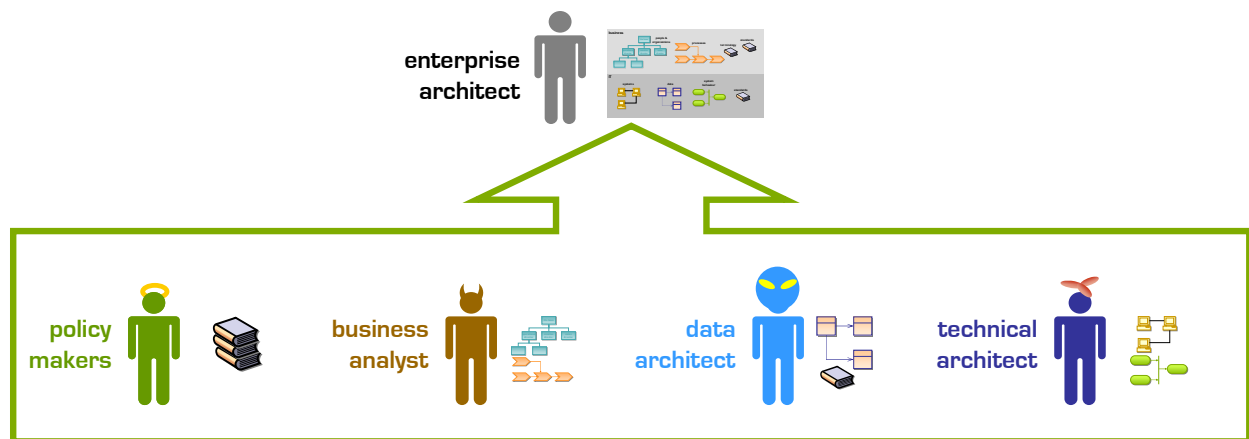
<sup>1</sup> see “The Value of Enterprise Architecture” by Tony Brown – [www.zifa.com](http://www.zifa.com) – the paper also provides other insights into the tangible benefits of enterprise architecture. This is also available for download on [www.modaf.com/Documents](http://www.modaf.com/Documents)

## Enterprise Architecture Models

There are two key types of Enterprise Architecture model; *as-is* and *to-be*. As-is models are generally used to understand the dependencies in an enterprise, or provide a “big picture” executive summary. To-be models are generally produced to examine the impact of change, or to act as a blue-print for a particular change programme. Quite often, to-be models will be developed to overlap a small part of a larger as-is model – for example to examine the broader impact of a change to a part of the enterprise. These hybrid models are sometimes called “transitional architectures”, because they support changes in the enterprise.

## The Enterprise Architect

A glance at a substantial Enterprise Architecture model would lead the casual observer to believe that the architect was in some way omnipotent. It would appear that one person was sufficiently skilled to model everything from business strategy, through processes and information models, to IT hardware compatibility. Thankfully, the Enterprise Architect does not need to be an expert in all these domains, but he or she must be sufficiently versed in their terminology to be able to communicate unambiguously with the owners of the information needed to build the architecture:



If the Enterprise Architect needs one skill above all others, it is persuasiveness – especially on first-time EA projects where there will be a certain amount of scepticism to counter. He or she will spend most of the time trawling the organization for sources of information to complete the jigsaw of the architecture model. The powers of persuasion are initially needed just to get the custodians of the information to share it. Then they are needed again to persuade those same custodians that actually a more up to date or accurate model is really needed. Finally, the real snake-charming skills are required to persuade the custodians to help keep the EA model up to date by communicating changes as and when they occur. In many ways, this is the perfect EA end-state, where all the stakeholders are bought-in to the idea and see the benefit of keeping it up to date. Generally, this is a far better way to run an EA programme than a dictatorial approach, but inevitably some stakeholders will be reluctant to co-operate and may need some coercion.

There is no single background or qualification that one should look for in an Enterprise Architect. Possible candidates may have started in IS/IT and been promoted to a position where they understand how the business functions at a high level. Equally, someone who has worked in an operational role but had to interface with IT suppliers may be equally suitable. The key requirement is for a generalist who understands the enterprise (usually this means they should be recruited from within) and has some IT experience. Communication and presentation skills are essential, because the results of the EA analysis are often used to support decisions made at the highest levels of the business. It is this rare ability to tease a concise and understandable view of a complex problem that is the mark of a true Enterprise Architect.

## EA Enablers

Given the breadth of information covered by an Enterprise Architecture Model, it is sensible to have a formal way of categorising the information it represents – i.e. a framework. The best known architecture framework was devised by John Zachman<sup>2</sup>, and defines a set of views on the enterprise. These views are designed to serve the needs of various stakeholders – e.g. business process models, logical data models, etc. Since Zachman, there have been a plethora of frameworks, emanating from national governments, IT consortia and businesses. Most owe their existence to Zachman though. The table below summarises the Zachman Framework<sup>TM</sup>:

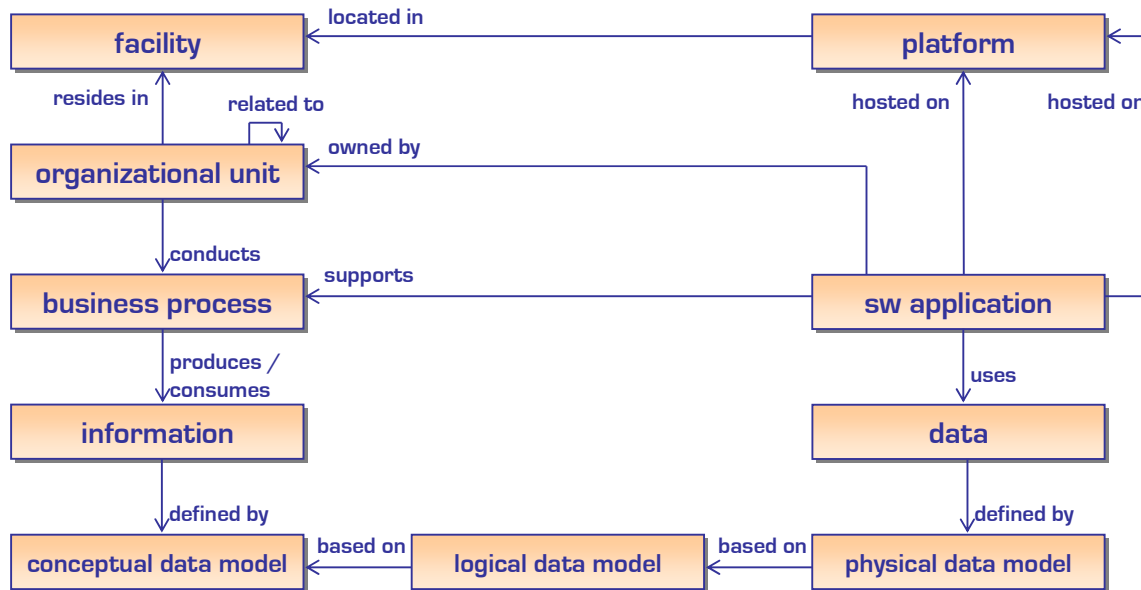
	DATA (what)	FUNCTIO N (how)	NETWOR K (where)	PEOPL E (who)	TIME (when)	MOTIVATIO N (why)
SCOPE (contextual)						
BUSINESS MODEL (conceptual)						
SYSTEM MODEL (logical)						
TECHNOLOGY MODEL (physical)						
DETAILED REPRESENTATI ON						
FUNCTIONING ENTERPRISE						

It is easy to be seduced by the managerial simplicity of an architecture framework. However, it is far more important to have a complete and contiguous model of the enterprise so that proper analysis can be conducted. The views defined by a framework are only really useful in presenting certain aspects of the complete enterprise model to specific stakeholders, or in specifying the information required from various parts of the enterprise. It should also be noted that not all views are useful, and great care should be taken not to produce views on the enterprise that have no business justification. Also, standard frameworks can never anticipate the questions that businesses will seek to answer through EA, so there will inevitably be a requirement for new or hybrid views in certain cases.

As the practise of EA matures, it is becoming obvious that producing a set of diagrams conforming to an architecture framework is of limited use. It is possible to use these models to answer a particular question, but re-use of disconnected diagrams is difficult. It is more useful to have a fully-integrated architecture model which can be kept up-to-date and re-used from project to project. To do this, a special type of enabling technology is required – a repository. Repositories allow the architect to import models from various parts of the organizations and integrate those models to produce a coherent model. First generation repository tools were not initially aimed at enterprise architects and simply provided a means to consolidate and configuration-control models. The recent generation of repositories have introduced querying and reporting functionality to better enable decision support, and are aimed squarely at enterprise architects.

Using a repository is all very well, but it will only answer the questions you ask if it has been properly configured. The key to successful repository usage is an architecture meta-model. A meta-model is, put simply, a model of a model – in other words, it is an information model describing the kinds of architectural elements used in the repository and the relationships that can exist between those elements. For example, one might want to assert that people and organizations conduct business processes (this relates the org structure to the process models), or that an organization is based in a building or facility. The simplified meta-model below shows the kinds of elements and relationships one might expect a repository to manage:

<sup>2</sup> see [www.zifa.com](http://www.zifa.com)



It is becoming common for architecture frameworks to be underpinned by a meta-model. These models vary from simple taxonomies of element types (see the US Federal Enterprise Architecture Programme) to fully-attributed logical models (see the UK MOD Architecture Framework - MODAF Meta Model).

## Summary

Enterprise Architecture closes the gap between business and IT. It provides a way for management to understand the consequences of their decisions and allows simple questions to be asked about complex organizations. It is difficult to measure specific cost benefits of an EA approach, but figures are starting to emerge that show how IT disasters can be averted by closing the gap between business and IT.

EA is not an IT discipline - a good Enterprise Architect possesses a rare fusion of business, technical and communication skills. Great care should be taken in selecting a candidate to fill an EA role, and the higher levels of management should be fully behind the architect, especially as the EA programme is starting up.

As the EA market grows, there have been an increasing number of modelling tools and repositories designed (or adapted) to serve the enterprise architect. Some of these tools can offer great benefits in terms of architecture re-use and ability to query the architectural model. A great deal of thought must be put into the configuration of these tools, as re-configuration can be costly when there is a large back-catalogue of architectural data.

## About the Author

Ian Bailey has worked on a number enterprise architecture framework projects for government and commercial organizations. He is responsible for the UK MOD's MODAF meta-model and provided technical expertise to the MODAF Partners delivery team. He is also the lead modeller on the IDEAS initiative<sup>3</sup>. Previous to this, he was the editor of the ISO10303-233 systems engineering standard, and an early contributor to the SysML specification.

for more information, contact [info @ modelfutures.com](mailto:info@modelfutures.com)

<sup>3</sup> [www.ideasgroup.org](http://www.ideasgroup.org)