White Paper

Does it matter which Architecture Framework we use?

WP0080 | June 2013

Roger Evernden

Roger Evernden has been an Enterprise Architect since 1984, specializing in the highly practical use of EA to manage enterprise transformation.

He acts as advisor, mentor and coach on enterprise architecture initiatives, leads training workshops, and writes regularly about strategy and architecture. His work has been the basis for more than 400 business and IT architecture initiatives worldwide. As author of the Information FrameWork (IFW) – an architecture framework originally developed for the financial services sector – Roger pioneered many contemporary techniques, including the use of industry reference models, business capability analysis, and component-based architecture building blocks.

Architecture Frameworks lie at the heart of EA work, but there remains a great deal of confusion about the subject. While different architects prefer one framework over another they don't often make their selection criteria explicit.

Why and how do we choose an architecture framework? And does it matter which one we choose and use? In this paper I will explain why frameworks are important, show that the framework we choose does indeed have an impact on our success, provide some criteria to make a selection, and suggest that we need to use more than one at a time!

Zachman or TOGAF®, or ...

Should Enterprise Architects use the Zachman Framework, or The Open Group Architecture Framework (TOGAF), or the Pragmatic Enterprise Architecture Framework (PEAF), or Information FrameWork (IFW), or .... Well this list could go on and on! If you search on the Internet you will even find sites that have catalogued the available architecture frameworks. I found one claim that there were more than 900 architecture frameworks, although more considered lists put the number at around 50[i].

Clients often find the Zachman Framework too complicated and theoretical. One common complaint about it is that it doesn't include detailed guidelines for using it in architectural process. A major flaw is that Zachman devised the framework in the late 1970s and early 1980s at a time when standalone, mainframe systems were the norm. Although

Zachman has issued newer versions, the Zachman Framework has significant limitations for the sophistication of contemporary EA.

TOGAF does provide detailed process guidelines in the ADM, and it is arguably the most detailed framework currently available, but it also attracts a great deal of criticism, not least because it remains largely IT-centric.

Obviously there are one or two contenders that are more popular and well known, but even so, how do you select between one and the others? And more importantly, does it matter which one you choose?

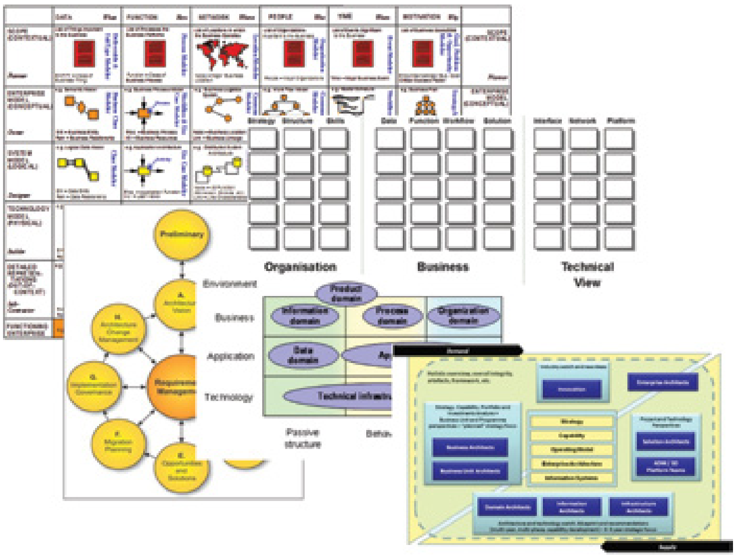


Figure 1: Graphics from some of the many frameworks

In this article I'm going to provide some simple guidelines to explain why an architecture framework is so important, and to give you some criteria for choosing one. I'll also provide some examples to show that it does matter which one you choose! And finally I'll explain why you probably need more than one framework.

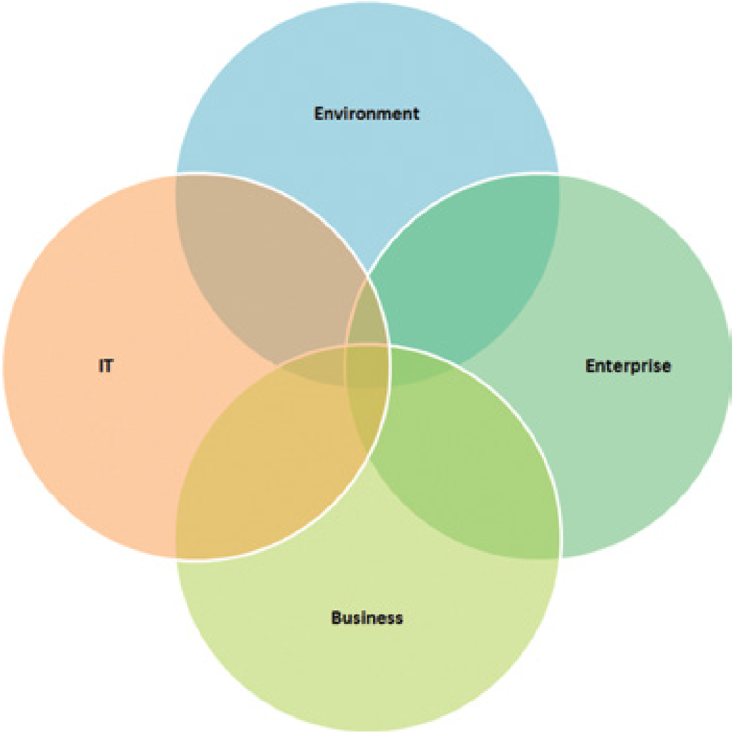


Figure 2: The vast scope of EA

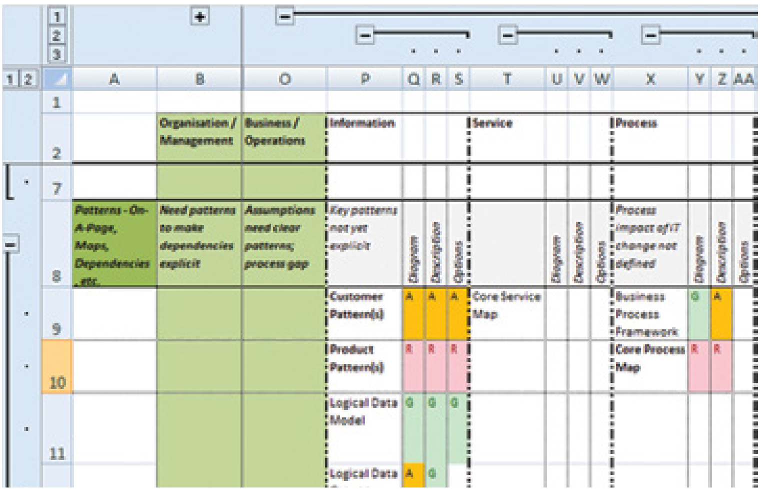
Why is an architecture framework so important?

There is some confusion over whether an architecture framework is necessary or not. Some people would even argue that you don't need one. So let me start by emphasizing that an architecture framework is one of the most important, practical and useful tools available to an Enterprise Architect.

Let me start by defining an architecture framework. According to ISO, an architecture framework *"establishes a common practice for creating, interpreting, analyzing and using architecture descriptions within a particular domain of application or stakeholder community"*. Some frameworks, such as TOGAF, cover a lot of ground. But from a practical perspective a framework like TOGAF is more a huge body of knowledge about architecting. In fact, TOGAF is more a framework of frameworks! If it is to be useful, a good framework must be a practical tool that helps an architect in their day-to-day work.

There are pros and cons for every architecture framework. Getting one that works for you is vital to your success as an architect! So start by being clear about what you need to achieve.

You wouldn't expect each project to have exactly the same project plan. In the same way you will need to create an architecture framework that is tailored to your exact needs.



	A	B	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1															
2		Organisation / Management	Business / Operations	Information				Service				Process			
7															
8	*Patterns - On-A-Page, Maps, Dependencies ..etc.*	*Need patterns to make dependencies explicit*	*Assumptions need clear patterns; process gap*	*Key patterns not yet explicit*	Diagram	Description	Options		Diagram	Description	Options	*Process impact of IT change not defined*	Diagram	Description	Options
9				Customer Pattern(s)	A	A	A	Core Service Map				Business Process Framework	G	A	
10				Product Pattern(s)	R	R	R					Core Process Map	R	R	
11				Logical Data Model	G	G	G								
				Logical Data	A	G									

Figure 3: An example part of a framework as a dashboard

Enterprise architecture is a vast discipline that covers a huge area. It needs to consider the environment in which an enterprise operates, to include the needs of management, leadership and decision makers, and to respond to the needs of business managers and operations, and the needs of IT and systems. EA also needs to consider the relationships and dependencies between all of these components, as a whole and from many alternative views and viewpoints. Then it has to create a future vision of what all of this might be like in the long-term, consider alternative possibilities, and come up with detailed roadmaps to explain exactly how to transform and change these complex structures in a managed and optimal progression!

A good architecture framework keeps track of this vastness and complexity by providing a high-level outline of all of the factors that are considered by the EA team. Just as program and project plans keep track of resources, tasks and milestones, and architecture framework is used to focus and govern everything that falls within the remit of the Enterprise Architect. In this respect, a framework is a tool for governance and management within the EA team.

As well as providing the conceptual skeleton for fleshing out details of the architecture, a framework serves as a dashboard by including measures and metrics to track what has been done, what needs to be done, and how well EA is meeting its objectives. In this way, a framework can also provide detailed metrics for use within the EA team, and a simplified dashboard for effectively communicating summarized information with stakeholders and reporting to senior management.

Criteria for choosing an architecture framework

As a framework is such an important tool for governing the enterprise architecture and for communication and metrics, how should we choose one framework over another?

Broadly speaking there are two types of criteria to consider. The first is about what a framework covers and what it doesn't cover. The second is to do with the usability and usefulness of a framework.

What a framework covers and what it doesn't cover

If you were to examine all of the frameworks available, you would find that they all covered certain factors in common. There are several academic and practical studies that have identified and listed these common factors, but their conclusions are fairly consistent. Several years ago I summarized these findings as a list of eight factors, which have become popularly known as the Evernden Eight!

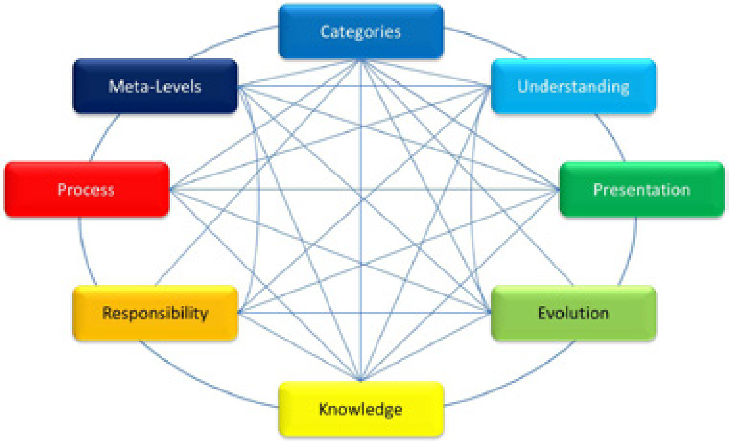


Figure 4: The Eight Factors found in all Frameworks

Briefly – every architecture framework covers one or more of these eight factors. For example, all frameworks divide the architectural scope into categories, the most common high-level categories being business, data, application and technology. Some frameworks describe the process of architecting, the most notable being TOGAF which describes the Architecture Development Method (ADM). All frameworks refer to the evolution of an architecture – from current, through transition, to the future, but not all of these explicitly include evolution as a dimension of the framework.

Let me give you a couple of examples of organizations using frameworks that didn't cover all of the factors that they needed:

- For many years one organization failed to consider Meta Levels and didn't produce an architectural metamodel. This caused a lot of confusion. For example, there was no agreement on what constituted an "application" - opinions varied from a "system" of multiple applications, to "modules", to an instance of executable code. It was only by thinking about and including meta levels that they avoided the constant misunderstandings.
- Another organization didn't make a distinction between architectural understanding and solution understanding (in TOGAF this is partly the different thinking between Phases A to D and the other phases, and partly the distinction between the architecture and solution continuums). As a result they were unable to get support for the architecture team, which was eventually disbanded!

How do you decide whether a framework covers the right things? The starting point is always in knowing what you want to achieve. Why do you need a framework? It might be that you need a simple, one-factor framework that shows the high-level categories and sub-categories that you will be using. Or you may need a framework that shows how each category is captured and stored in a repository – using a matrix between categories and meta-levels. Or one that shows who is responsible for each step in the architectural governance process – using a combination of process and responsibility.

Selecting an architecture framework using these eight factors is therefore a question of:

1. Being clear what it is that you need to manage – why you need a framework.
2. Deciding what types of information will help you – which of the eight factors are relevant.
3. Deciding how you combine the eight factors to form a framework.

Note: this might result in more than one framework! I'll come back to this in a moment.

Usability and usefulness of a framework

The second set of criteria for choosing frameworks is more to do with how useful and practical it is.

You are more likely to use this second set of standards if you are choosing one of the many pre-defined frameworks. If you create your own frameworks using the eight factors that I described earlier, then you will already have taken usability and usefulness into account.

You don't need many criteria. Figure 4 has a basic set that I use with my clients. A framework has to be **relevant** – it must have a specific, stated purpose. It must make sense (be **sensible**) for governance of EA and for communication with stakeholders.

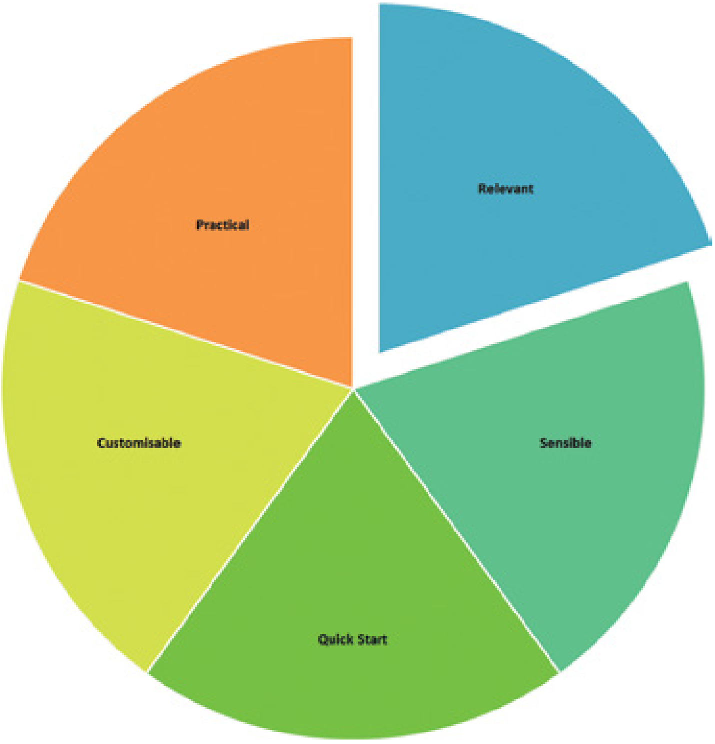


Figure 5: Simple Criteria

It must allow a **quick start** – it needs to provide just-in-time support, without requiring a lot of effort to set it up. If there are checklists and materials that you can use to define the framework – that makes it easier and quicker to get started. For example, I use checklists for each of the eight factors so it is simply a question of choosing the elements that are needed for each dimension of a new framework. TOGAF has a lot of material that can be adapted without too much effort.

It should be **customizable** – because no two enterprises are the same, your needs will change over time, and you will need to adapt the framework as you go. Also no two enterprises are the same, so whatever you choose it will need to be tailored to your specifics. Additionally, your needs will change over time, so your frameworks will need to adapt as circumstances change.

Finally, it needs to be **practical**. It should be a must-have, day-to-day tool that makes your job easier and better. If it doesn't do that, then you've got the wrong framework!

Do you need more than one framework?

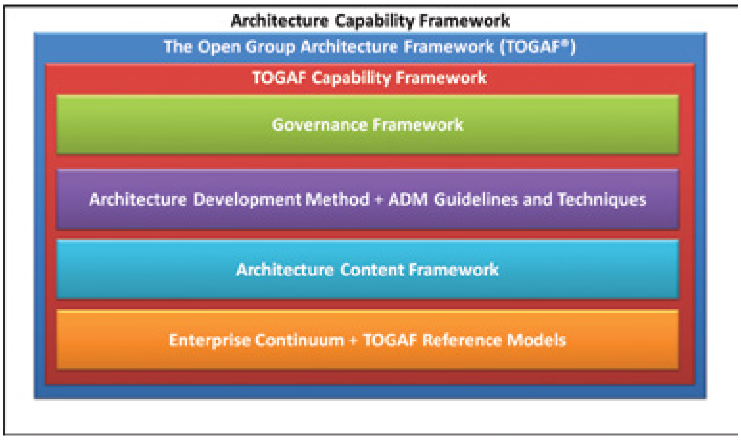

Architecture Capability Framework
The Open Group Architecture Framework (TOGAF®)
TOGAF Capability Framework
Governance Framework
Architecture Development Method + ADM Guidelines and Techniques
Architecture Content Framework
Enterprise Continuum + TOGAF Reference Models

Figure 6: TOGAF® - framework of frameworks

Earlier I hinted that most architects need more than one framework. Isn't this going to make life more complicated? So firstly let me repeat that a good framework must be a practical tool that helps an architect in their day-to-day work! Although TOGAF is a useful body of knowledge that provides excellent best practice source material, it isn't a good framework in this sense. As I said earlier, TOGAF is more a framework of frameworks. The practical bits of TOGAF are the individual frameworks within the broader body of knowledge.

So if you are using TOGAF, you are already using more than one framework!

There are two reasons why you need more than one framework:

- If you include too many factors in a single framework it becomes too complicated and difficult to use. The Zachman framework is an example of a framework that tries to cover many different factors in a single diagram.
- Using several frameworks allows you to apply the right tool for each job. The Architecture Content Framework is about deciding and governing architectural information. The Enterprise Continuum is about integrating reference models with your organization-specific architectures.

Software support for EA has improved dramatically, and some of these tools, including iServer, support multiple, customizable frameworks. As suggested in this paper, to carry out the EA role effectively requires more than one framework, and each framework must be tailored to your needs. Tooling is vital in supporting application of a framework.

Conclusion

Frameworks are probably the single most practical and useful technique available to architects.

Unfortunately they are sometimes seen as theoretical, cumbersome or irrelevant in our day-to-day work. This is because we need to choose the right framework for our needs. To do this we need to start by thinking about what we want to achieve, and then deciding what types of framework will effectively support us.

If you follow the guidelines in this paper you will indeed discover that frameworks are practical, simple and relevant on a daily basis.

[i] See the ongoing survey of architecture frameworks at:
http://www.iso-architecture.org/ieee-1471/afs/frameworks-table.html

Orbus Software
3rd Floor
111 Buckingham Palace Road
London
SW1W 0SR
United Kingdom

+44 (0) 870 991 1851
enquiries@orbussoftware.com
www.orbussoftware.com