

White Paper

2nd Iteration

Practical Considerations When Using
TOGAF® for the Second Time

WP0095 | August 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.

Much of the writing on TOGAF® is about starting out in Enterprise Architecture. For example, the Introduction to TOGAF in Part 1 of the TOGAF documentation “provides an executive overview of enterprise architecture, the basic concepts of what it is (not just another name for IT Architecture), and why it is needed. It provides a summary of the benefits of establishing an enterprise architecture and adopting TOGAF to achieve that”. Explaining the benefits of EA in the first place is all very well, but we also need to think about the issues organizations face when they continue to use TOGAF!

Iteration is simply the repetition of a process such as the ADM. But although TOGAF describes Applying Iteration to the ADM in Chapter 19, it doesn't really highlight the huge difference between a first and second iteration of ADM.

Why does it matter? Well most of all – we all want to be successful. We want to know that we are doing something worthwhile and, deep down, that we still have a job to go to. But underlying the need to survive are some considerations that often get overlooked. I know of too many organizations that completed the first iteration of TOGAF without really considering how things have changed for the second iteration. One company cancelled their EA programme – despite a highly successful initial iteration – because the EA team couldn't cope with escalating demands. In a way they were victims of their own success, but looked at another way they might have prevented their early demise by being better prepared.

Access our **free**, extensive library at
www.orbussoftware.com/community

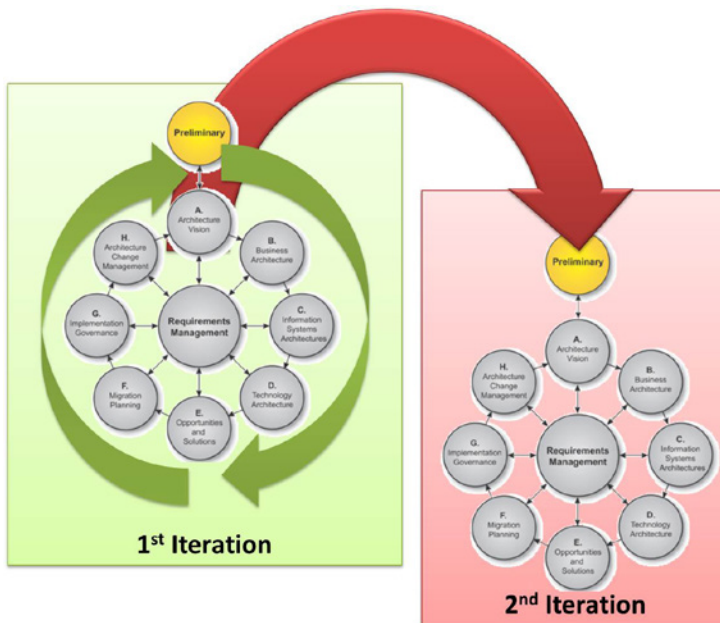


Figure 1: Keep revisiting the Preliminary Phase during the 1st Iteration to be fully prepared for the 2nd Iteration

TIP: Start thinking about the 2nd iteration of TOGAF as early as possible. Don't leave it too late to adequately prepare. Figure 1 is a reminder to keep the Preliminary Phase in mind throughout the 1st Iteration in order to be best prepared for the 2nd Iteration.

Typically there are three BIG changes between the first and second iteration:

1. The number of stakeholders increases. And with this, requirements get more complicated, and the EA role of balancing trade-offs, priorities and politics comes to the fore.
2. The role of EA artefacts, frameworks and repositories (eg, iServer) becomes more crucial. These were all important in the first iteration, but now they become really critical success factors.
3. The EA development process becomes even less sequential. In some enterprises the process can be quite chaotic!

The Number of Stakeholders Increases

The first iteration is often undertaken as a “proof of concept”. Even with organizations that have made a big commitment to EA, initial work goes on in parallel with recruiting and training, establishing procedures and governance, installing software and repositories, and fitting in with existing teams and politics.

At a bank, because the focus of the new EA team was on renewing the Internet platform for the retail division they were working with a relatively small number of stakeholders from one area of the bank. After 8 months, as part of the next strategy planning cycle, the team suddenly found that they had new stakeholders covering the corporate division, small to medium business, insurance, wealth planning, and overseas business. In addition, retail stakeholders expanded to include mortgages, personal loans, account opening, and savings, as well as the ongoing work on the Internet platform.

At other organizations the number of stakeholders expands because of greater involvement in strategy and investment decision making, closer collaboration with IT development and operations, or the need to deliver greater operational efficiencies.

In every case, during the 2nd iteration requirements get more complicated and the role of EA in balancing trade-offs, priorities and politics is more critical.

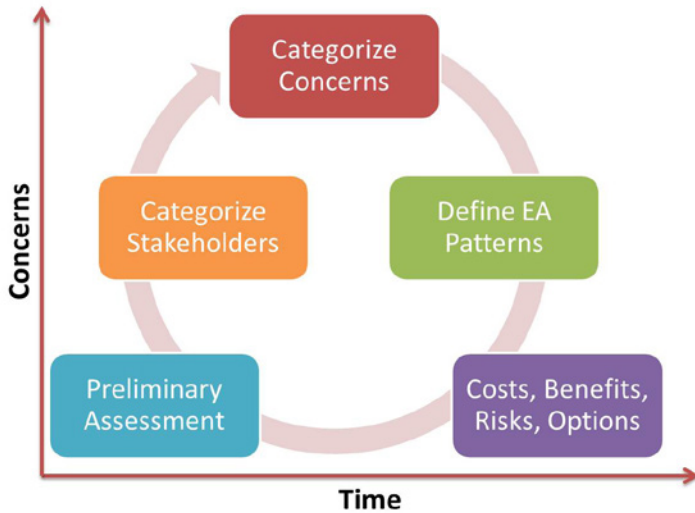


Figure 2: Managing Stakeholder Concerns

TIP: Map stakeholders and their concerns onto architecturally meaningful patterns.

Here are my top tips:

- Consider all stakeholders as individuals, but then categorize and group them to make communications and deliverables as simple as possible. TOGAF provides some guidelines to help do this (for example, in 24.3.1.1 Sample Stakeholder Analysis). Then categorize their concerns into related areas of concern.
- Define clear future architectural options. By the time you come to the second iteration, stakeholders are more likely to want a range of ideas and suggestions. Map each set of concerns to relevant EA patterns. TOGAF covers Architecture Patterns in Chapter 25. Be prepared to explain and compare each alternative. Include metrics and criteria to help stakeholders choose between them. At the very least you need to include the costs, benefits, risks, and future options, from a business and executive perspective, for each architectural transition.
- Above all, work with stakeholders by including them in your architectural dialogues. Remember EA is all about addressing their concerns, so engage with them when gathering or providing information, in bounded and open dialogue, and during consultation.

Figure 2 shows this process over time. Generally speaking, you will include more stakeholders on the 2nd and subsequent iterations of TOGAF. As the number and complexity of concerns increases, the architectural role is made easier the more effectively you can group concerns by mapping them to EA patterns with related costs, benefits, risks and options. In effect, over time this cycle helps to manage stakeholders and concerns by dealing with them as a smaller number of architecturally meaningful chunks!

The Role of EA artefacts, Frameworks and Repositories becomes more crucial

It is rare that repositories and frameworks are well-defined by the first iteration. And most organizations start EA with very few pre-defined or reusable artefacts. So during the first iteration, these things are still being worked out.

In effect, the Preliminary Phase of TOGAF is an on-going set of tasks. They don't just happen at the outset of an EA programme and then get forgotten. Instead, think of the Preliminary Phase as the things you need to do constantly in order to achieve the highest level of EA maturity! Essentially the Preliminary Phase is a complete misnomer – it should be the Optimizing Phase, covering Level 5 of the maturity levels as it is about “continuous improvement of the enterprise architecture process”.

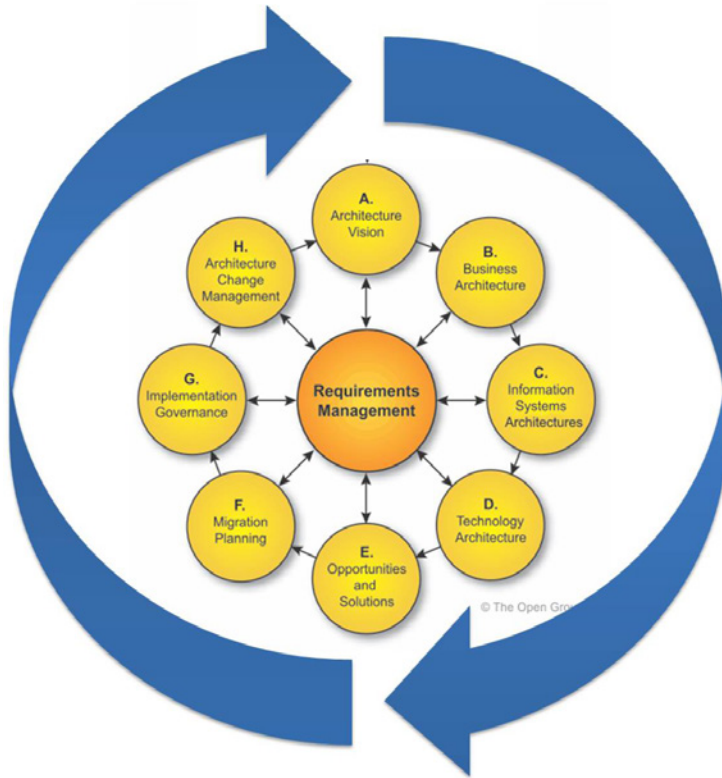


Figure 3: The Optimizing Phase

What does this mean when embarking on the 2nd Iteration of TOGAF? Let me give you an example. When a pharmaceutical company started their first iteration of TOGAF, they hadn't fully decided what EA repository or software they would use. So throughout the first iteration they used a mishmash of PowerPoint, Visio, Spreadsheets, Word, and SharePoint (Orbus Software's iServer for Visio would have helped greatly here). The EA team became engrossed in getting the first iteration of TOGAF right, so they followed the sequence of Phases in ADM with little thought about the Preliminary Phase. There was considerable and growing interest in using EA by the time they were heading to the 2nd iteration – with each iteration roughly coinciding with the annual planning and investment cycle. With increasing pressure to provide EA services there was significantly more demand on limited EA resources, and the frameworks and “repository” remained in the primitive state they were in at the outset of the first iteration. Because they had to follow the internal acquisition process it was a further 9 months before they began installation of dedicated EA software, and a further four months before this was fully operational in a live environment. Because they hadn't considered the tailoring of the framework to match the metamodel of the software, it was well into the fourth year before the team made any effective use of the EA tool.

So here are my top recommendations to help make the 2nd iteration more successful:

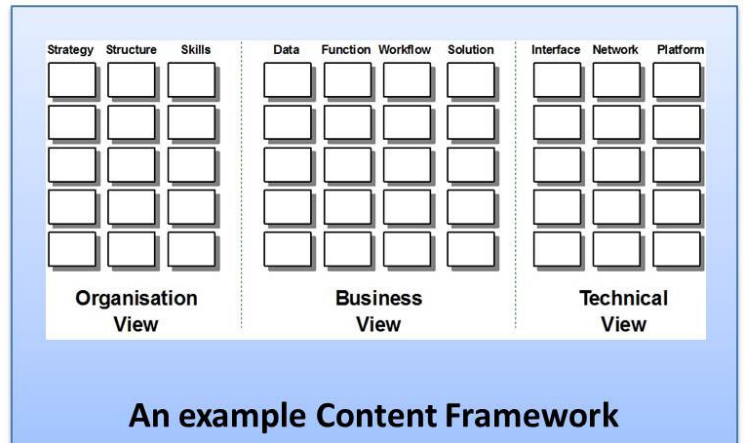
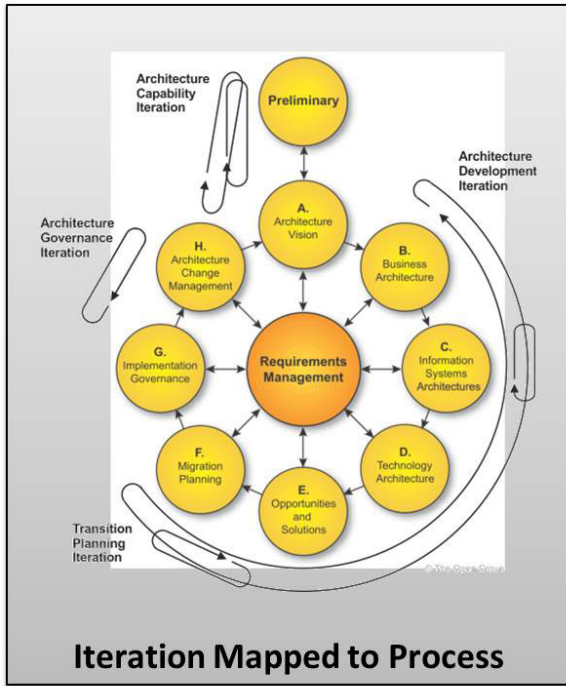
- Don't wait until the 2nd Iteration to repeat the "Preliminary" Phase. Constantly review the EA development process, and in particular how it could be improved through the better use of artefacts, frameworks and repositories.
- Remember that a "framework" is a practical, proactive EA management tool. It is not an obscure, theoretical chart that can be ignored. TOGAF presents itself as a best practice development process and a number of frameworks – governance, content, etc. But it doesn't give enough guidance on the practical use of these frameworks, and it gives no guidance on how to adapt frameworks to the specific needs of your enterprise. So find out how to really apply frameworks as tools to manage the on-going, day-to-day EA work. And do this throughout the first iteration so that you have the stamina for the 2nd iteration!
- And bear in mind that a repository supports the EA frameworks, not the other way around. Define the EA frameworks you need to manage the concerns, requirements, artefacts, and other EA deliverables. Frameworks provide the criteria to help you select the appropriate software and repositories. You need software that is flexible enough to support your frameworks, especially the content and governance frameworks.

We come now to the third, and possibly biggest, difference between the 1st and 2nd iterations.

The EA Development Process Becomes even less Sequential

In my introduction I said that in some enterprises the process can be quite chaotic! The problem is partly that the TOGAF ADM is never used in a purely sequential way – there is always some iteration between Phases, as acknowledged by the TOGAF discussions on iteration. This problem is then compounded because each project is likely to be using a different part of the ADM at different times, and that there are dependencies between projects, and between Enterprise, Segment and Capability partitions of the overall architecture.

All of this is recognized in the TOGAF documentation, but TOGAF overlooks the easiest way to handle the iterative development of EA – which is to map deliverables to a dynamic, interactive scorecard. A simple way to do this is to design a Content Framework that provides a high-level overview of all deliverables. Figure 4 shows these two iteration mapping options – mapping iteration to the ADM process, and using a Content Framework as the basis for a management scorecard.



Using the scorecard option immediately simplifies the iterative nature of architectural development. Instead of trying to keep track of each iteration against the ADM crop circle diagram, a content framework is used to track the availability, use and reuse of deliverables. The TOGAF view of iteration is iteration of process, whereas architectural value is delivered through the use of appropriate artefacts!

What is the best way to create a Content Framework?

- On one dimension it should cover all of the domains within EA scope. This should include the TOGAF set of business, information systems (data and application) and technology. The key is to separate out each concern as a distinct domain, or column, in the framework.
- The second dimension should make the distinction between Architecture and Solution, which is a key part of the Enterprise Continuum described in TOGAF. In the Content Framework this distinction is important to separate architectural from solution deliverables, and to be able to track how architecture artefacts are reused as patterns and templates across multiple solutions.
- For each cell there should be measurable criteria that are used to track deliverables that are required and available.

P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	
Information				Service				Process				Event				Information Technology	Applications & Interface				Network & Security				Infrastructure				
		Words	Definition	Classification	Link between services and capabilities	Words	Definition	Classification	Link from process to application needs work	Words	Definition	Classification	Business as usual - not really covered	Words	Definition	Classification	Clearer standards for language and	Knowledge needs to be explicit and in Trust	Words	Definition	Classification		Words	Definition	Classification	Needs to be more explicit	Words	Definition	Classification
Product Taxonomy	R	R	R	Service Catalogue	G	G	G	Activity Hierarchy	G	G	G	Trigger Hierarchy	G	A	G		Application Taxonom	A	A	A	Channel / Device taxonom	A	R	R	Infrastructure Taxonom	A	R	R	
Data Classifications	G	G	G	Function Model	A	A	A																						
Principles (data)	A	A	R	Principles (service)	G	A	A										Principles (applicati Architectural	A	A	R	Principles (network	A	A	A	Principles (infra-structure	A	A	A	
Key patterns not yet explicit	Diagram	Description	Options		Diagram	Description	Options	Process impact of IT change not defined	Diagram	Description	Options	N/A	Diagram	Description	Options	Patterns emerging - could have more	Detail good-patterns would give consistency	Diagram	Description	Options		Diagram	Description	Options	Consistent reference architecture as needed	Diagram	Description	Options	
Customer Pattern(s)	A	A	A	Core Service Map				Business Process Framework	G	A							Application Architecture Pattern(s)	R	R	R					Technical Reference	R	R	R	
Product Pattern(s)	R	R	R					Core Process Map	R	R							Application Map	G	G	G					IT Product Catalogue	A	A	A	
Logical Data Model	G	G	G														Application	G	G	G									

Figure 4: Example of an Interactive Content Framework

Using a Content Framework it is easy to see what has been created by the architecture and where it is being used. A good Content Framework allows top-down, bottom-up, iterative and ping-pong development of the architecture! One simple way to create an interactive content framework is to use a spreadsheet (see Figure 5), with columns or rows for the different types of domain information, and separate cells for each metric or measure. This is an approach that I've used with many organizations, as it allows simple sharing of a scorecard, and can be easily summarized for communications with key stakeholders.

Conclusions

Obviously each iteration of TOGAF is going to be different from previous ones, but the 2nd is nearly always a significant departure from the 1st. The differences are often overlooked, but thinking about the three BIG changes discussed in this paper will help you cope better with the demands of the 2nd iteration!

© Copyright 2013 Orbus Software. All rights reserved.

No part of this publication may be reproduced, resold, stored in a retrieval system, or distributed in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the copyright owner.

Such requests for permission or any other comments relating to the material contained in this document may be submitted to: marketing@orbussoftware.com

Orbus Software

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

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

