



An Enterprise Architecture Development Framework

Adrian Grigoriu

Trafford Publishing

2007

\$49.95 227 pages

Reviewed by Paul Harmon

Each year there are a number of Enterprise Architecture conferences. If you attend one, you will probably find that the attendees discuss an architecture from two rather different perspectives. The larger group comes from an IT background and is focused on defining and organizing IT resources. This group thinks that “Enterprise Architecture” is really just a synonym of an “IT Architecture” and that both, in essence, involve cataloging and tracking IT resources. The second group consists of people who are concerned with figuring out how all of the elements in an enterprise fit together. These people may come from a Business Process Management perspective, in which case they will emphasize the importance of the Business Process Architecture as a key element in any comprehensive Enterprise Architecture. Others, from slightly different perspectives will focus on different elements within the enterprise, and most who are not IT oriented will leave the conference thinking that they won’t be coming next year, as the conference is “too techie.”

What is true of Enterprise Architecture conferences is equally true of Enterprise Architecture books. There are a few that take a broad perspective, but most focus on creating an IT architecture. *An Enterprise Architecture Development Framework*, by Adrian Grigoriu, is a pleasant exception. Grigoriu takes a very comprehensive view of things and works to show how everything can be integrated within a broadly conceived view of things.

He begins by considering the benefits to be derived from an architecture, and then proceeds to discussing how to cost various benefits and how to build a business case for an Enterprise Architecture effort. The heart of the book is an explanation of Grigoriu’s Functions*, Flows, Layers and Views (FFLV) Enterprise Architecture Framework. In essence, the FFLV framework is a three dimensional cube that classifies Functions, Flows, Layers and Views and shows how they integrate to define all of the elements in an enterprise. His Layers, for example, are Business Processes, Technology and People. Once Grigoriu has defined his model, he works through a wide variety of technologies and standards to show how they fit into his model. He continues by classifying other Enterprise Architectures and shows where his fits among the alternatives. Finally, he spends quite a bit of time suggesting how one might approach creating and using an Enterprise Architecture.

I found Grigoriu’s FFLV Framework a bit too complex. As Grigoriu notes, quoting Zachman, in fact, a graphic is needed and useful because it provides a way for everyone to see how everything fits together. I agree, but fear that Grigoriu’s graphic is a bit too complex. In fact, one can hardly figure out what’s involved or how it fits by looking at the basic FFLV “cube.” To even begin to understand the elements involved in the “cube,” you will need to work your way through a longish chapter where he dissects the various layers and views and shows what is on each.

This isn't a model that would allow a user to identify a specific activity – say a “customer request”- - and see what is related to that activity. Grigoriu could certainly create a set of layers and views and explore the interconnections between the “customer request” activity and everything else in the enterprise, but it would involve a rather tedious walk through the diagrams of the various layers and views. In other words, Grigoriu's model is an abstract and very complex view of how all of the elements in an enterprise fit together. It is now a diagram that one might tailor to a specific company to show how that particular organization's elements work together.

The nature of Grigoriu's model leads directly to problems when he turns to describing how to go about creating an enterprise. He is describing an abstract process and not explaining how to accomplish specific tasks. Thus, for example, we gather that we should document the processes and flows in an organization, but aren't given specific models for documenting the processes and flows. We don't start with any concrete idea of what we are trying to create. This is a book that could really have benefited by a couple of good case studies to focus the discussion on how these ideas might be applied at an actual company.

On the other hand, Grigoriu has clearly surveyed a very large number of different technologies, and most of the popular approaches to Enterprise Architecture development. Thus, for example, he has a section where he describes the uses and fit of BPM, BPEL, EAI, SOA, ERP, SCOR, eTOM, ITIL, CIBIT, Value Chains, Six Sigma, CMM, MDA, etc. (Unfortunately Grigoriu has very few footnotes and no bibliography. He describes each of these technologies in a very different way but provides no guidance to learning more about them. If you already know them, it's interesting to see how Grigoriu fits them together with other technologies. If you don't know them you will soon be dizzy trying to keep track of all the acronyms.) I particularly liked his classification of EA frameworks: Matrices (Zachman), Cubes and pyramids (FEAF, BPTrends pyramid, and FFLV), Reference models (FEA), Functional architectures* – “in truth process maps” (SCOR, eTOM), best practice models, and combination models, like TOGAF. As he notes, most limit their scope to IT, but others are quite comprehensive.

I enjoyed reading this book. It was as if Grigoriu had laid out all the business and IT elements that make up an enterprise on a table and then played around until he could fit them all together into a single cube – an ingenious effort at puzzle solving. I wouldn't try to teach this model to someone who was new to architecture and wanted to do something practical, like create a process architecture or organize IT resources. On the other hand, I would happily recommend this book to anyone who is already an experienced enterprise architect. This book is an excellent “graduate review” that will force you to think through lots of issues and consider how you might address them in your own architecture.

I would also recommend this book to someone who was interested in the issues involved in building a business case for an Enterprise Architecture effort – the sections on benefits and costs are excellent and comprehensive – and I'd also recommend this book to someone who wanted to learn more about how to classify stakeholders. The section on strategy and stakeholders is outstanding.

Paul Harmon is the Executive Editor of Business Process Trends (www.bptrends.com). He is a recognized BPM analyst and consultant and the author of *Business Process Change*.

(* Note that Grigoriu uses “functional” in a slightly unusual way throughout the book and readers will need to make a special effort to avoid thinking of functional as “departmental” and allow it to include “processes.”)