White Paper

The Role of Views and Viewpoints in Enterprise Architecture Design

Moving beyond Catalogues, Matrices and Diagrams

WP0040 | August 2012

Rowan Napier

Rowan is an Enterprise Architect with Row1 Enterprise Engineering where he believes in "Architecting Interesting ThingsTM". Rowan is focused on the delivery of a business-appropriate and sustainable EA practice and offers EA Related and General Management "Asset-Light" Consulting services through 3rd party relationships and client engagements.

Rowan has consulted to clients in different industries in different roles and across architecture domains for 10 years. This varied experience coupled with his education in both business and information technology spheres has provided the basis for his enterprise architecture expertise.

He welcomes your comments. at:
rowan@row1enterpriseengineering.com
https://www.linkedin.com/in/row1rowan

"Requirements error costs are high and fixing a requirements error after delivery may cost up to 100 times more than the cost of fixing an implementation error"

(Ian Sommerville, the well-known author of Software Engineering).

Adopting an Enterprise Architecture approach can potentially reduce the number of requirements errors because it is the continuous practice of describing the essential elements of a socio-technical organization, their relationships to each other and to the environment, in order to understand complexity and manage change within an organization (EA Research Forum definition of Enterprise Architecture). If you do not understand the organizational environment where a system is used, the system is less likely to meet the real needs of the business and its users.

A critical step in any Enterprise Architecture (EA) approach is the identification of stakeholders that are concerned with managing change within a socio-technical organization. The concerns of these stakeholders are addressed through the development of Enterprise Architecture products. TOGAF 9.1 refers to views that can take the form of diagrams, matrices or catalogues.

In this white paper I will propose the use of an Architecture Customer Development Process that can be used to validate source traceability, in other words create the link from requirements to the stakeholders who proposed them. I am a firm believer in the old adage that says "people will support what they help to create". That is why it is so important to tie in the human aspect of enterprise integration to the change methodology

and requirements process where people fulfil the role of change agent and that of potential and actual resource within the organization.

Stakeholders must validate the architecture products produced to meet their viewpoint specification and provide the intended rationale for the requirement. Thus EA products requirements validation is concerned with checks for validity, consistency, completeness, realism and verifiability by other humans.

Using GERAM as Validation Reference

How do you develop and validate your Enterprise Architecture product requirements as part of your Enterprise Architecture Methodology?

Start by using a reference architecture and methodology as benchmark. I propose the Generalised Enterprise Reference Architecture and Methodology (GERAM).

GERAM describes an important aspect of enterprise architecture and engineering by recognizing and identifying feedback loops on various levels of enterprise performance as they relate to its products, mission and meaning. To achieve such feedback with respect to both the internal and the external environment, performance indicators and evaluation criteria of the corresponding impact of change on process and organization are required. The continuous use of these feedback loops will be the prerequisite for the continuous improvement process of the enterprise operation and its adaptation to the changes in the relevant market, technology and society.

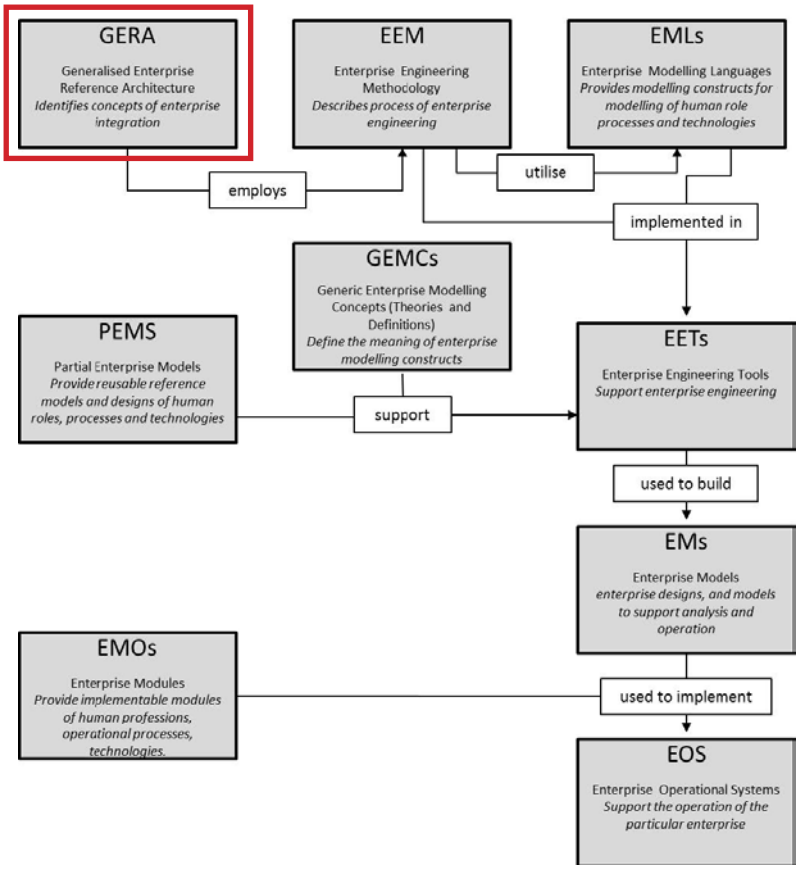

Figure 1: GERAM - Framework for Enterprise Engineering And Enterprise Integration

Figure 1 is a representation of all the components of GERAM and the GERA component defines the enterprise related generic concepts recommended for use in enterprise engineering and integration projects. These concepts can be categorized as:

a) Human oriented concepts to describe the role of humans as an integral part of the organization and operation of an enterprise and to support humans during enterprise design, construction and change.

b) Process oriented concepts for the description of the business processes of the enterprise;

c) Technology oriented concepts for the description of the business process supporting technology involved in both enterprise operation and enterprise engineering efforts (modelling and model use support).

The concepts defined within GERAM contain details that are architecturally significant and leave the details that do not matter on the enterprise level for designers on a solution level.

Leonard Fehskens, VP, Skills and Capabilities at The Open Group positioned the importance of Enterprise Architecture in linking strategy with execution in his presentation, Deriving execution from Strategy: Architecture and the Enterprise. The key points are listed in figure 2 below.

Deriving Execution from Strategy: Architecture and the Enterprise

In a presentation written by Leonard Fehskens, VP, Skills and Capabilities at The Open Group titled "Deriving Execution from Strategy: Architecture and the Enterprise" Leonard makes two statements about the definition of an architecture:

- "Those properties of a mission, its solution and their environment that are necessary and sufficient for a solution to be fit for purpose for its mission in that environment."
- "An architecture defines a class of (acceptably equivalent) solutions that are fit for purpose for a class of missions in a class of environments."

Mr Fehskens then further qualifies what he understands the meaning of those two statements to be. He describers then as follows:

- "Architecture is about constraining decision-making options; it is about the things that have to be done a particular way to ensure that a solution is fit for purpose for its mission in those environments where it may be deployed."
- "Thus, there are details that don't matter, and they are not the subject of architecture. Downstream designers and implementers are free to make decisions about these details as long as these decisions are consistent with the architecture."

Figure 2: Extract from presentation delivered by Len Fehskens

Mr Fehskens finishes his article with some comments on the Architecture of Architecture concluding that there is a continuous and dynamic alignment required between an enterprise's mission, solution and environment. To him, what matters is the fit(ness) for purpose and essentials (the necessary and sufficient properties) of the architecture and the need for a continuous chain of decision making principles that provide motivation and justification for execution decisions, and models that express how these principles integrate with one another.

So with the context of human oriented fit(ness) for purpose in mind I am of the view that a formal Architecture Customer Development approach be adopted by the Enterprise Architecture organization to support and balance this continuous and dynamic alignment between an enterprise's mission, solution and environment.

Adopting a Customer Development Approach

In support of this line of thinking, Eric Ries the author of "The Lean Startup – How Constant Innovation Creates Radically Successful Businesses" relates the following story; "The business and marketing functions of a start-up should be considered as important as engineering and product development and therefore deserve an equally rigorous methodology to guide them."

He calls it the Customer Development Methodology

Other influential sources that mention the importance of managing customers' expectations and requirements are TOGAF 9.1 in figure 3 and Kimball's Data warehousing in figure 4.

TOGAF 9.1 Reference to Stakeholders / Customers of EA

Architecture Customer Development talks directly to the objectives of the Preliminary Phase in TOGAF 9.1 , specifically the Preliminary Phase (Chapter 6 section 6.2.2) Organizational Context:

- The stakeholders for architecture in the enterprise; their key issues and concerns.
- Current processes that support execution of change and operation of the enterprise, including the structure of the process and also the level of rigor and formality applied within the organization. Areas for focus should include:
    - Current methods for architecture description

Figure 3: TOGAF 9.1 Stakeholder / Customer Reference

Ralph Kimball's Reference to

Ralph Kimball makes use of a publishing metaphor from the publishing industry to express the responsibilities of the Data Warehouse and Business Intelligence organization as:

- Know your business users and decisions they make
- "Publish the right data" from a variety of sources that is:
  - Relevant
  - Understandable
  - Acceptable Performance
  - Minimizes cycle time (initial and on-going delivery)
  - Current methods for architecture description
  - Minimizes total cost of ownership
- Ensure data/application accuracy and quality
- Adapt to changing needs and realities
- Success determined by business (readers/subscribers)

Figure 4: Ralph Kimball's EDW & BI Team Responisibilties

Mr Ries goes on further to state that "the goal of a start-up is figure out the right thing to build-the thing customers want and will pay for- as quickly as possible." And that "this is achieved through validated learning, a process of demonstrating empirically that a team has discovered valuable truths about a start-up's present and future prospects" (See figure 5 below)

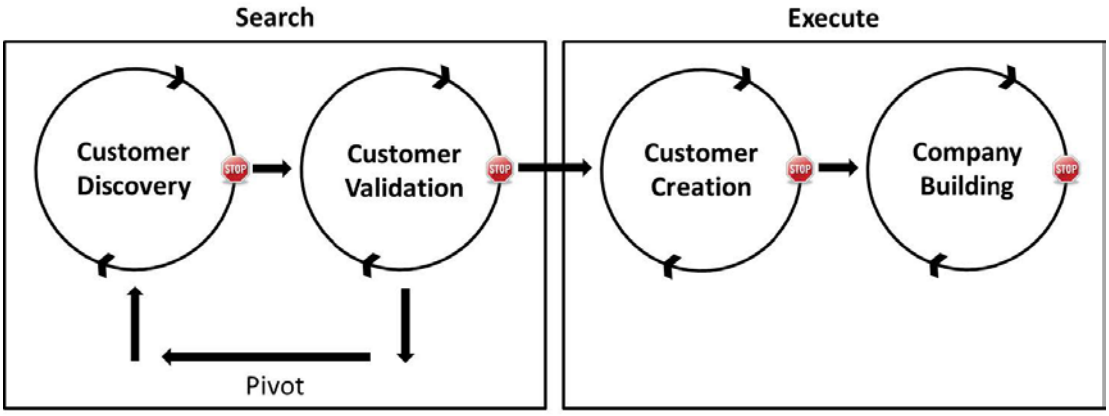

Figure 5: Customer Development Process

Another technique Mr Ries discusses as being crucial to continuous alignment is that of small batch sizes, where "the biggest advantage of working in small batches is that quality problems can be identified much sooner."

Richard Rumelt the author of the book "Good Strategy Bad Strategy: The difference and why it matters" discusses a process of learning as "hypothesis, data, anomaly, new hypothesis, data, and so on-is called scientific induction and is a critical element to every successful business"

Using a Architecture Customer Development Process to Manage Requirements

The customers of Enterprise Architecture can't always tell us what they want or what would be helpful to them and usually express requirements in their own terms.

So I feel that likening the Enterprise Architecture Organization to a start-up, considering the rapidly changing environments, the increasingly shortening time to show customer value and the role that experimentation plays in both environments to determine or elicit and validate customer requirements has merit.

The Enterprise Architecture Organization has a similar challenge to understand which of its activities are value creating and those that are wasteful.

Considering Enterprise Architecture and the **ISO 42010 "Conceptual Model of Architectural Description"** is analogous to the publishing, human oriented design and customer development metaphors mentioned above in that coherent architectural descriptions provide the rationale that addresses the stakeholders concerns and is in line with the objectives of the Preliminary Phase.

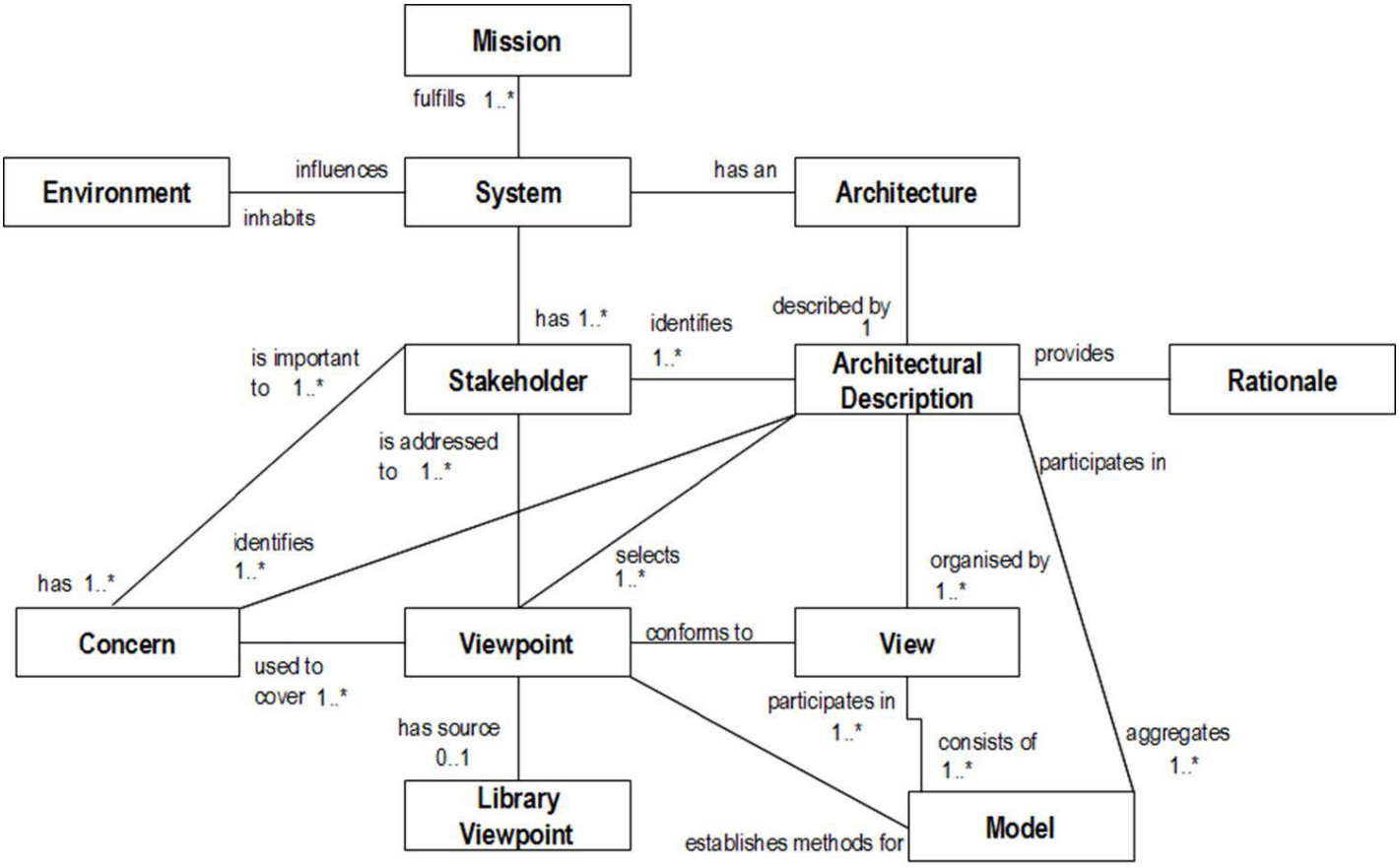

Figure 6: ISO 42010

What I suggest is that a more formal **Architecture Customer Development Process** as a *requirements elicitation and validation technique* be tied to the Requirements Management Process to ensure source **traceability** i.e. links from requirements to the stakeholders who proposed them and that by following this Customer Development Process, the enterprise architecture products delivered are of the right quality and support the current strategy of the organization in a format digestible by its stakeholders.

The iterative activities in the **Search Phase** highlighted in figure 5 "Customer Development Process" can be adapted for Architecture in the following ways:

1. **Customer Discovery** – the tasks in this process would be similar to the objectives of identifying the stakeholders and their concerns in the architecture effort but then to actually develop a plan to test our stakeholder's reactions to prototyped architectural descriptions in addressing their concerns.

2. **Customer Validation** – the tasks in this process would be to actually measure and evaluate the response of the stakeholders to our prototyped architectural descriptions (experiments) and return to Customer Discovery until the views in the architectural descriptions put forward to address stakeholders concerns are "necessary and sufficient for a solution to be fit for purpose for its mission in that environment"

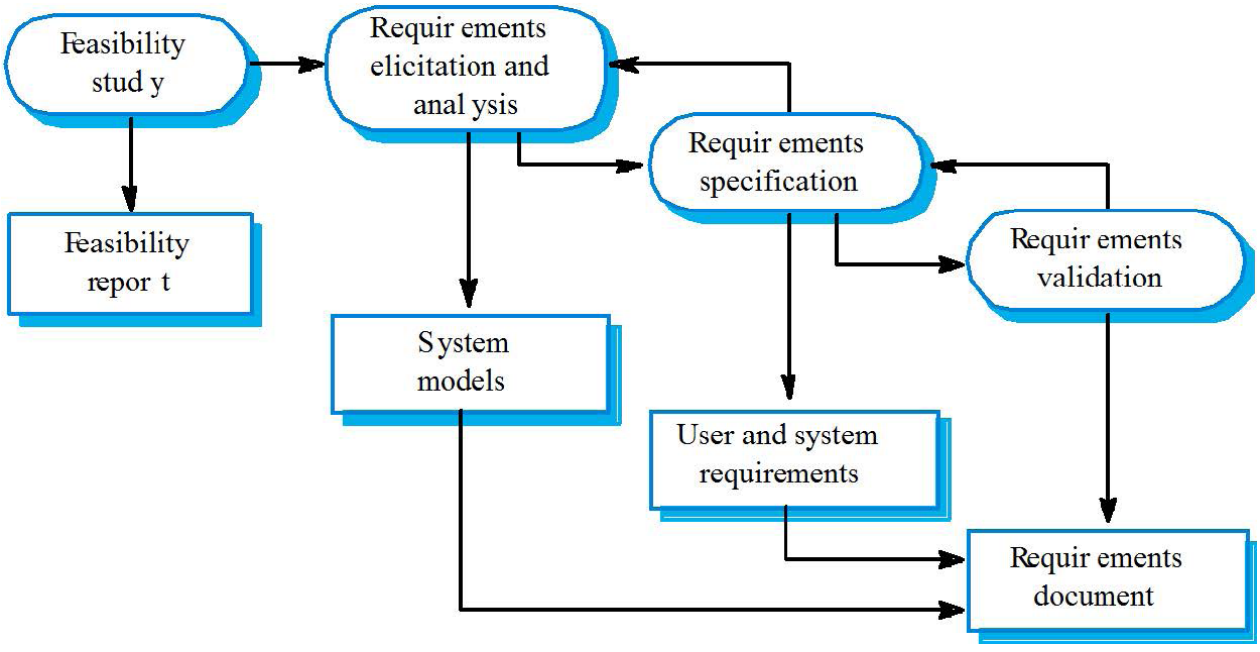

Figure 7: A requirements engineering process

These outputs would be fed into the Architecture Context Iteration to define or update the Content Framework and Meta-Model of the enterprise, the Execute Phase would be replaced and follow the rest of TOGAF ADM in terms of an Architecture Definition Iteration.

In my experience the lack of validated architecture descriptions and large architecture definition batch sizes causes a disconnect between the Enterprise Architecture Organization and the outputs required by Architecture Stakeholders in formats that allows them to fulfil their rationale for the work in the first place. I have also found this to be the reason for why Enterprise Architecture repositories fail to stay relevant and maintained in line with the realities of the enterprise.

Conclusion

While the concepts and relations between views, viewpoints and models are well known and understood tying these back to customer validated architectural descriptions that support the appropriate rationale must be improved and run within a formal Architecture Customer Development Methodology. The rationale must also have been validated in experiments with stakeholders before any architectural definition takes place and the Architecture Customer Development Methodology must facilitate requirements source traceability as a “continuous chain of decision making principles that provide motivation and justification for execution decisions, and models that express how these principles integrate with one another.”

The idea behind the Architecture Customer Development Process is to prototype the individual architectural descriptions including their participating views and supporting text to the architecture stakeholders. This is achieved by following a customer development methodology to validate the level of abstraction required, the format for delivery to support the idealised use thereof by the stakeholder, whether it be a presentation, natural language documentation, published content online, a poster or a mix of text and graphics, and that the Architectural Description combining a collection of viewpoint purposes supports the rationale of the stakeholder(s).

An Architecture Customer Development Process can be aligned to the Requirements Management Process of developing and evolving an organization’s Enterprise Architecture within the context of the Architecture Context Iteration of TOGAF.

This identified viewpoint mix across stakeholders and their concerns can then be used to formulate the most optimal design with which to create and or update your enterprise’s content framework and content meta-model. This must then be implemented within an Enterprise Engineering Tool such as Orbus’s iServer which leads to the topic for a future paper.

References

1. Generalised Enterprise Reference Architecture and Methodology, Version 1.6.3, IFIP–IFAC Task Force on Architectures for Enterprise Integration March 1999
2. EARF - *http://hufee.meraka.org.za/Hufeesite/collaborations/earf/definition-for-ea-as-defined-by-the-group*
3. Ian Sommerville - Software Engineering (7th Edition)
4. Leonard Fehskens – The Open Group
5. Ralph Kimball – Dimensional Modelling In-Depth Course, 30Aug – 2nd Sept 2011
6. Eric Ries - The Lean Startup – How Constant Innovation Creates Radically Successful Businesses
7. Steve Blank and Bob Dorf – The Start-up Owner's Manual – Customer Development Process Fig 2.1
8. Richard Rumelt - Good Strategy Bad Strategy: The difference and why it matters
9. TOGAF - *http://pubs.opengroup.org/architecture/togaf9-doc/arch/*

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

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

orbus software