Experiences of Tool-Based Enterprise Modelling as Part of Architectural Change Management

Conference Paper · January 2015

DOI: 10.5220/0005887902550260

2 authors:

Vaughan Michell
University of Reading

Cameron Spence
University of Reading

ISSN 2166-6768 (print)
ISSN 2166-6792 (online)

Association of Enterprise Architects®

Journal of Enterprise Architecture

2016, Volume 12, Number 3

Journal of Enterprise Architecture®

About the Journal: The Journal of Enterprise Architecture (JEA) is published by the Association of Enterprise Architects, 44 Montgomery Street, Suite 960, San Francisco, CA 94104, Tel: +1 415 374-8280, Fax: +1 415 374-8293, www.globalaea.org. The JEA is a peer-reviewed international publication for the EA community. Four issues are published each year. The JEA supports the global practitioner community of interest through the publication of articles that promote the profession of enterprise architecture, and deals with issues regarding practices and methods, case studies, and standards at the national and international levels. The views expressed in JEA articles are those of the respective authors, and not necessarily those of the publisher, the Chief Editor, the associate editors, or the Association of Enterprise Architects (AEA).

Article Submissions: Authors may submit properly formatted manuscripts to the JEA Chief Editor at journal.submissions@globalaea.org for publication consideration. Author submission guidelines are available on the AEA website at www.globalaea.org. Copyright of all accepted articles and other published content is transferred by the author to the JEA upon notification of acceptance by the JEA Chief Editor.

Subscription: Available to AEA members. The annual cost of AEA membership is US$75.00. To join, please refer to www.globalaea.org. Online edition: no charge access for AEA members.

Back Issues: All back issues of the JEA are available in electronic form, and can be accessed online by AEA members. Libraries can apply for access by contacting the AEA.

Contents

From the Editor

Leonard Fehskens

I am pleased to introduce this issue by calling attention to a change on the Journal's masthead – the appointment of Chris Forde to the position of CEO of the Association of Enterprise Architects, succeeding Allen Brown who has retired. I have known and collaborated with Chris as a respected colleague for many years, especially as successive Forum Directors of the Architecture Forum of The Open Group, and look forward to his tenure as the AEA CEO. I also thank Allen for his enthusiastic and unflagging support of the Journal as an AEA asset.

We open this issue with a brief report of a survey conducted by McKinsey and Company and the Henley Business School.

Two articles that address the application of enterprise architecture to specific business categories in Portugal (health) and Australia (retail) follow.

Then we have a short subject on a set of principles to facilitate an architectural approach to "whole of enterprise" issues, and another that takes a "big picture" view of the implications of the "digital mesh" for enterprise architecture.

Change and transformation remain on the tips of everyone's tongues, and so we have an article that develops an idea that seems, as we used to say in college, "intuitively obvious to the most casual observer", except for the fact that this is the first time I have seen it articulated – that the ability to change is itself a business capability and should be thought of and developed as such.

The issue closes with the sort of thing that the EA community aspires to but rarely achieves – a mathematically grounded technique for assessing the quality of models.

Len.

Visit the AEA website at www.globalaea.org

Call for Papers

The Journal of Enterprise Architecture is accepting article submissions for its future issues. Research and best practice articles are sought on enterprise architecture-related topics, including:

- Case Studies, Configuration Management, Culture, Documentation
- Evaluation, Frameworks, Governance, Implementation, Maintenance
- Methodologies, Taxonomies, Theory, Training, Tools, Use, Value

Please send articles to the JEA Chief Editor at journal.submissions@globalaea.org. Author submission guidelines can be found on the AEA website at www.globalaea.org.

Article

EA Survey Findings: The Challenges and Responses for Enterprise Architects in the Digital Age

Sharm Manwani and Oliver Bossert

Abstract

Enterprise architects face many challenges to be relevant to key stakeholders. The growth of digital business offers major opportunities for enterprise architects if these challenges can be addressed. The Enterprise Architecture (EA) Survey created by McKinsey & Company and Henley Business School explores EA outcomes and capabilities to assess the responses to the challenges. This article highlights key findings from the survey as a call to action for EA leaders.

Keywords

Enterprise architecture role, digital, stakeholders, survey

INTRODUCTION

The challenges enterprise architects face are illustrated by the Forbes article, "Is EA Completely Broken?". This states that: "misdirected Enterprise Architecture (EA) initiatives vastly outnumber *bona fide* examples of EA efforts leading to measurable business value".

Yet, arguably with the advent of disruptive digital technologies, EA is even more critical to help redesign business and operating models. Digital business has helped IT evolve from the back office to the front line, but connecting the dots is not easy. Increasingly, digital business capabilities can create value by allowing a company to create new online products, establish an omnichannel strategy, run analytics, or increase process automation.

In order to contribute to the digital transformation, enterprise architects must ensure that organizations can manage the complexity of legacy systems while enabling the innovation critical to remaining competitive – particularly where there are new digital business entrants. Some experts argue that this requires a two-speed architecture (Bossert 2014) given that stability and agility are not easy goals to reconcile.

The intellectual demands of EA are high, but arguably stakeholder engagement is even more challenging. For EA to proactively influence strategy, communications skills are pivotal, requiring language targeted to the boardroom. The role of a business architect has become important in this context, as evidenced in an informal survey organized by Allen Brown, former CEO of The Open Group. We conclude that EA is most effective if it also participates in the formulation of and engages in the execution of strategy. That means working closely with or potentially acting as solution architects in major transformation programs.

The above challenges and responses were key elements of the EA Survey.

EA SURVEY APPROACH

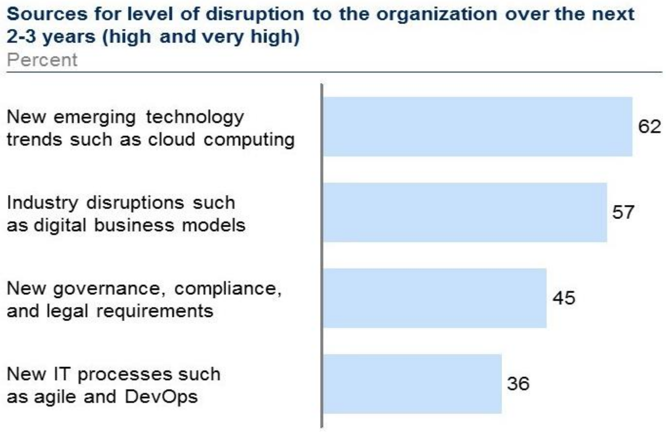
The EA Survey covers EA outcomes and capabilities and targets CIOs, EA heads, and business leaders. It is an online survey with to-date 100+ respondents. The authors reported results at the McKinsey CIO Conference in February 2016. This enabled selected panel experts to comment on the results and the audience to give feedback. Here we share some survey highlights.

EA SURVEY FINDINGS

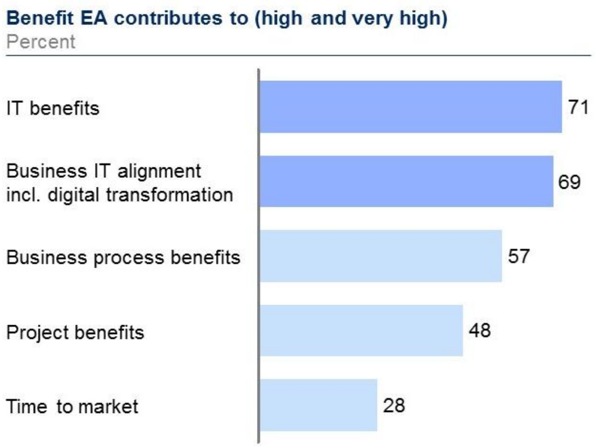
We were able to build a solid fact base of the "typical" setup of a large EA department. Respondents were mainly from large companies with approximately 3,000 FTE IT staff. EA-dedicated resource (internal/external) is about 1% of Total IT FTE, supplemented by 2.5% outside of EA. Two out of three EA groups report to the CIO/CTO.

EA for the Digital Enterprise

We did not just look for a description of EA – we wanted to understand the value it delivers. New digital business models are expected to be one of the major disruptions following the opportunities that new emerging technologies offer.



EA is key to digital transformation contributing significant benefit to business IT alignment.



EA seems to play an effective role in digital transformation since, in companies where EA contributes, we found that projects are more likely to be delivered on time and budget.

Implementing Digital in a Two-Speed Architecture

In a two-speed architecture, EA has to ensure agility as well as a stable foundation and smooth integration. Time to market is the key driver for agile use and is critical for success in digital.

How important is an agile approach in meeting outcomes?

We use agile development mainly for fast-moving applications	49.50%
We use agile development for most projects	27.23%
We do not use agile development	23.27%
Total	**100.00%**

The survey found that most companies are using agile but with different approaches. About half the companies are using agile mainly for fast moving applications. The remainder, fairly evenly split, are at opposite ends mostly using agile or not at all.

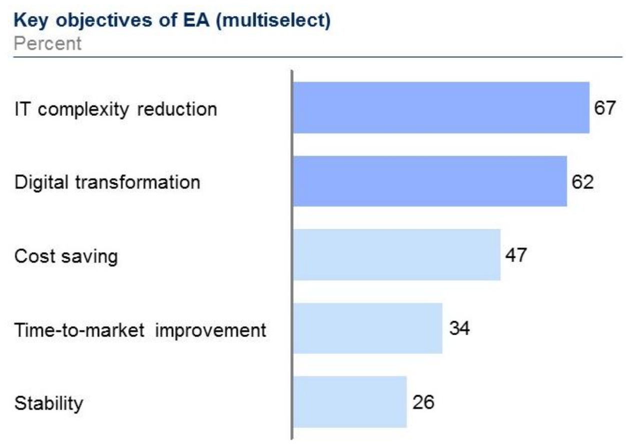
The interaction of EA with agile also varies between organizations. More than half of the companies have a different EA approach for agile development. More than two-thirds of initiatives use a tailored accelerated EA management process for agile, while the others have separate EA guidelines for agile. Yet 43% of the companies do not change their EA approach at all for agile projects.

There is some support for the conflicting goals of agility and stability. Companies with digital business high on their agenda typically have more point-to-point connections, a lower quality of business process documentation, and less reuse of services.

How to Manage Complexity in a Fast-Moving Environment

Complexity in the IT landscape has to be addressed, particularly if the goal is to be agile and innovative.

While digital transformation is a key objective of EA in most companies, complexity reduction is seen as at least as important as digitization.



Yet there is a high spend on integration at 40% of total application development spend. Part of this is due to the high number of interfaces relative to the number of applications. This finding is further reinforced by comparing organizations with lower *versus* higher integration costs.

	Lower than average integration cost	vs.	Higher than average integration cost
Percent of point-to-point connections	17		62
Number of applications	245		694
Number of interfaces	522		2,242

Enhancing Communication and Leadership of EA

Perhaps the biggest challenge for enterprise architects (and unfortunately somewhat supportive of the Forbes finding) is that in more than 40% of companies, business colleagues were unaware of what EA does.

What is also a concern, although perhaps not too surprising, is that in these low EA awareness organizations it appears that EA is both more reactive and more rigid in its approach.

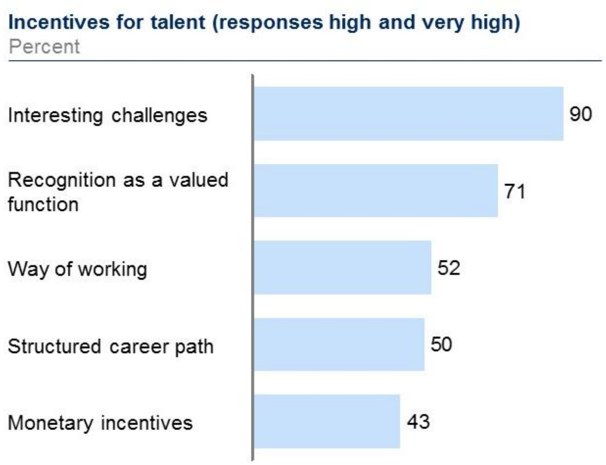
Specifically, we found in regards to capability that:

- EA does not help deliver business solutions, is not considered to be necessary for managing standards, and business architecture is less mature.
- There is often no capability or process model in place and no benefit capturing.
- EA lacks the right talent to cope with business and technology challenges.
- EA is less likely to model the future.

In relation to EA approaches, we found that:

- EA is more focused on compliance than on strategic planning.
- EA is more likely to use architecture methods by the book.
- EA typically runs agile projects with the same governance as other projects.

These findings emphasize the importance of leadership in attracting and retaining top EA talent. Here it seems that Herzberg's hygiene and motivating factors come into play with enterprise architects more incentivized by interesting challenges than money.



CONCLUSIONS AND NEXT STEPS

The McKinsey and Henley EA Survey aims to provide insights into key drivers of a successful EA strategy for the digital age. We believe relatively little data exists regarding EA performance in this context. The EA Survey should create a sound basis for tackling EA challenges and opportunities from new disruptive events. This can help EA leaders develop a perspective on what does and does not work well.

The initial results of the EA Survey suggest that EA is a key enabler for digital transformation, but that IT complexity and EA leadership are significant barriers to achieving this in most companies.

Digital business is both a key driver and major objective of EA in most companies. High complexity is evident – not helped by the number of application and interfaces despite the ongoing drive for "jigsaw" architectures implied by SOA and its many predecessors.

The focus on both agility and stability provides some support for a two-speed architecture although the concern that 40% of spend is on integration reinforces that this cannot be treated as an afterthought.

Last but certainly not least is the need for strong EA leadership both to engage with business colleagues and to attract top talent. This is vital in order for EA to be proactive in driving digital innovation.

NEXT ROUND OF EA SURVEY

The authors wish to express their thanks to all the AEA members who participated in the EA Survey and provided such an excellent set of data for our analysis. The next round of the EA Survey is now available. Please look out for communications from the AEA on the launch of this survey.

ABOUT THE AUTHORS

Sharm Manwani is Executive Professor of IT Leadership at Henley Business School, where he directs business IT programs for IT leaders, EA professionals, and consultants. He was previously European CIO and responsible for business architecture at Electrolux.

Oliver Bossert is a Senior Expert on Enterprise Architecture and Technology at McKinsey & Company. Within this role, he applies enterprise architecture to create and implement multichannel strategies for customer-facing organizations in a digital context.

REFERENCES

K. Dörner, D. Edelman: What "Digital" Really Means, McKinsey Quarterly, July 2015.

Agile Enterprise Architecture Finally Crosses the Chasm; refer to: www.forbes.com/sites/jasonbloomberg/2014/07/23/agile-enterprise-architecture-finally-crosses-the-chasm.

Is Enterprise Architecture Completely Broken?; refer to: www.forbes.com/sites/jasonbloomberg/2014/07/11/is-enterprise-architecture-completely-broken.

Frederick Hertzberg: One More Time: How do you Motivate Employees?, Harvard Business Review, 2003.

O. Bossert, C. Ip, J. Laartz: A Two-Speed IT Architecture for the Digital Enterprise, 2014; refer to: www.mckinsey.com/business-functions/business-technology/our-insights/a-two-speed-it-architecture-for-the-digital-enterprise.


ED&BI
Enterprise Data & BI
Conference Europe 2016
7-10 November 2016
Radisson Blu Portman Hotel
London
Benefits of Attending:
Five Conference Tracks with More Than 60 Sessions with a Focus on Case Studies. Learn from other organisations past successes and challenges from over 30 case studies. The five tracks include a one day Big Data track and a one day Data Innovation track as well as tracks on Business Intelligence and Enterprise Data.
Sixteen Half Day Workshops. Choose from an unparalleled range of workshops on specific topics to get you quickly up-to-speed or fine tune your performance on Data and BI Essentials. Choose from introductory or advanced levels.
Full Day Workshops. Choose from four full day conference workshops. Enhance your in-depth knowledge and skills on Enterprise Data Management and BI best practices.
Wide Variety of Topics Providing an Excellent Mix of Theory, Visions and Practical Implementations. Topics include Agile BI, Big Data Modelling, Data/Information Maturity Assessment, Data Quality, Logical Data Warehouse, Big Data Platform Fundamentals, SQL and NoSQL, Data Protection Regulation, Information Governance, Data Funding, Predictive Analytics, Self Service & Collaborative BI, Getting Business Value from the IoT, Big Data Technology and Use Cases, Data Virtualisation, Information Architecture, Enterprise Data Integration, Data Governance, MDM and many more....
Lt Col Mike Servaes CDO British Army
Sue Geuens Head: Data Standards & Best Practice Barclays & DAMA Int
Rhys David Head of Business Intelligence & IT Delivery Gocompare.com
Roberto Maranca Managing Director Enterprise Data GE Capital
Pablo Casullo BI Development & Strategy Lead Merck Sharp & Dohme
Rick van der Lans R20/Consultancy
Len Silverston Universal Data Models
William McKnight Consultant McKnight Consulting Group
Graeme Simsion International Best-Selling Novelist (and Former Data Modeller)
Mike Ferguson Intelligent Business Strategies
Case Studies & Contributors Include:
Gocompare.com
IFC World Bank
The British Army
Barclays
Merck Sharp & Dohme
Visa Europe
Royal Mail
Ford Motor
Nordea Bank
Environment Agency
Zurich UK General Insurance
City of Amsterdam
IBM
GE Capital
Dubai Customs
Volkswagen Pon Financial Services
Generali
Boehringer Ingelheim Pharma
Dell IT Enterprise Architecture
Yorkshire Building Society Group
Dublin City University
Friesland Campina
Premier Farnell
Computer Weekly
Cloudera
Talenom
Trifacta
Sberbank CZ
IDG
Zipcar Int
OulalaGames
Speakers Include:
Rick van der Lans
Sue Geuens
Mike Ferguson
Jan Henderyckx
Malcolm Chisholm
William McKnight
Chris Bradley
Len Silverston
Graeme Simsion
Mike Simons
Daragh O Brien
Nigel Turner
Jos van Dongen
Brian McKenna
Garry Manser
Gwen Thomas
Peter Thomas
Neil Storkey
Produced By:
IRM UK
Strategic Business & IT Training
irmuk.co.uk
In Association With:
DAMA International
Hosted By:
DAMA United Kingdom
irmuk.co.uk/edbi2016

Information Reference Architecture for the Portuguese Health Sector

André Vasconcelos and Tiago Brás

Abstract

The main goal of information architecture is to identify and define the main types of data that support an organization's business. Information architecture provides the description of the informational entities required for the pursuit of the organization's business processes. The information architecture aims to identify key information to the business, define the data independently of applications or systems, and provide the basis for the management of corporate data. In a more general way the existence of an Information Reference Architecture (IRA) guiding and restricting the instantiations of a group of architectures and individual solutions is indispensable. In this article we propose a method to develop an IRA in order to ensure easy maintenance and semantic interoperability, through a bottom-up approach that uses a group of Information Systems (IS) in a specific business category. It is a four step, bottom-up method that starts with the mapping of the main IS of a business category, and with reverse engineering, model enhancements, and model integration techniques enables the creation of an IRA. This method is used for proposing the IRA for the Portuguese Health Sector, culminating in the development of an IRA for that sector. We used the Design Science Research Methodology (DSRM) to conduct our research. The method proposed in this work and the corresponding instantiation to the Portuguese Health Sector are assessed with evaluation metrics.

Keywords

Healthcare Industry, Informational Entities, Information Reference Architecture, Interoperability, Model Integration

INTRODUCTION

In an ever more global and competitive world, organizations have become very dependent on information to implement their activities. Therefore, improving the creation, management, and exchange of information is fundamental to ensure a continued competitive edge and to increase global performance (Watson 2000).

The achievement of these improvements is based greatly on the development of Information Architectures (IA) that facilitate the sharing and exchanging of information and the easy and fast development of information services (Watson 2000).

When considering the variety of business categories, the difficulty in assuring continuous competitiveness is even higher, as it is necessary to take into account the large number of organizations and Information Systems (IS) that integrate a given sector. Assuring interoperability among all the organizations and IS is, quite often, dependent on the existence of an Information Reference Architecture (IRA) guiding and restricting the instantiations of a group of architectures and individual solutions and easing the maintenance of the different IS (CIO 2010). This article addresses the subject of IRA and suggests a method that, applying a bottom-up approach and starting from a set of IS, allows us to get to an IRA that can assure maintenance efficiency and semantic interoperability. To reach this method, we use some of the work done by Vasconcelos and Brás (2015). While that work assumes that the models of the analyzed IS were standardized and ready to be integrated, in this work we assume that it is necessary to carry out a standardization of these same models before the integration. As far as the Portuguese Public Administration (PA) is concerned, there is only an IRA for the whole PA (*Agência para a Modernização Administrativa* 2015). Different business categories integrating the Portuguese PA, as it is the case of the health sector, have no IRA. That is the reason for several semantic incompatibilities existing among the IS in the Portuguese Health Sector. These incompatibilities are responsible for increasing the difficulty in maintaining the IS involved.

After having developed the method suggested in this research, we proceed with its instantiation and validation in the Portuguese Health Sector.

The structure of this research follows the Design Science Research Methodology (DSRM) (Henver et al. 2004).

PROBLEM IDENTIFICATION AND MOTIVATION

Following the frame presented in the introduction, it is clear that: *There is no IRA for the health sector in*

Portugal. Generalizing this motivation, and having in mind that this work can be applied to other business categories, the problem to which this research intends to find an answer is the following:

There isn't a well-defined method for the creation of an IRA that follows a bottom-up approach and that assures an easy maintenance and semantic interoperability.

Research Questions

The main research question that guides this document is:

- How to develop an IRA that can assure easy maintenance and semantic interoperability?

In order to be able to answer this question, there are three other questions without which it is impossible to correctly orient this work:

- How are we going to compare the informational entities of different models?
- Which IRA will be adequate for the Portuguese Health Sector?
- Which informational entities ought to be present in the IRA for the Portuguese Health Sector?

Having the answers to the above research questions, we can, then, correctly address the problem identified for this work and develop a method that allows us to create an IRA that is easy to maintain.

RELATED WORK

In this section the most relevant concepts of the work related to the topic under consideration that can contribute to the resolution of the previously identified problem are analyzed.

Enterprise Architecture

Enterprise Architecture (EA) is a set of principles, methods, and models that are used in the design and implementation of the organizational structure of a company, its business processes, IS, and infrastructure (Lankhorst et al. 2005). An EA consists of several layers (Godinez et al. 2010). As far as the development of this work is concerned, we will focus on the informational layer of an EA, since this layer is responsible for creating and managing the IA.

Information Architecture

Information Architecture (IA) describes the principles and guidelines that allow a consistent implementation of information technology solutions (Godinez et al. 2010). Furthermore, it aims to identify and define the main types of data that support the organization's business (Spewak and Hill 1993).

IA brings many advantages, such as to:

- Facilitate the sharing and exchanging of information
- Improve security and privacy
- Respond more rapidly and effectively to customer demands
- Facilitate the integration of systems, processes, data, and information
- Increase the business understanding (Watson 2000)

An IA is composed of Informational Entities (IE), characterized by their attributes and relationships that the different IE establish among themselves (Vasconcelos 2001).

International Reference Architecture and Document Produced in the Context of the Portuguese Health Sector

Although this article is not restricted to a specific business category, an instantiation for the Portuguese Health Sector will be done in order to tackle the motivation identified in the *Problem* section. That's why it is useful to have in mind the work carried out in a similar context.

HL7 RIM

The Reference Information Model (RIM) is the fundamental support for the development of version 3 of the HL7 (Health Level Seven International, 2014). This model aims to represent all clinical information from HL7, and can be seen as a reference model for the Health Sector.

INTEROPERABILITY, REVERSE ENGINEERING, MODEL ENHANCEMENT, & MODEL INTEGRATION

Interoperability

The concept of interoperability refers to the ability of different heterogeneous applications to share processes and data in distinct systems or platforms (Bernstein 1996). To make this communication possible, it is necessary to take into account the existence of different types of interoperability – organizational, semantic, technical (Interoperability Solutions for European Public Administrations 2010).

Regarding the specificity of this work, we highlight the semantic interoperability, since it is the one that occurs in the informational layer of an EA and is related to the way different IA are represented.

Among the various ways identified to improve that interoperability, the reverse engineering, the model enhancements, and the model integration acquire great importance (Tran et al. 2008; Chiticariu 2008). In Vasconcelos and Brás (2015) the steps of reverse engineering and model enhancements are not taken into account because it is assumed that the models of the analyzed IS are standardized and ready to be integrated. However, in this article we consider that these steps should not be ignored.

Reverse Engineering

Reverse engineering is the process of analyzing a particular system in order to identify its components and the relationships between these components, in order to create representations of the system in another form or at a higher level of abstraction (Chikofsky and Cross 1990).

In Bagui and Earp (2003), a set of transformations used in reverse engineering at the database levels is presented.

Model Enhancements

Model enhancements help models to become more correct, simple, and understandable (Batini et al. 1986). Besides using good modeling practices, in Rizopoulos and McBrien (2009) and Rizopoulos (2009) some of the most common transformations in model enhancements are defined.

Model Integration

Model integration consists of building a global view, taking as its starting point a set of models developed independently (Rahm and Bernstein 2001). Since the development of such models occurs in an independent way, the structures and terminologies presented are often very different amongst themselves.

Therefore, it is necessary to understand how we can combine these differences in order to create a single and coherent model.

In Batini et al. (1986) the following steps in model integration are identified: pre-integration, models comparison, conforming models, and models unification.

In the pre-integration step it is necessary to analyze the models we wish to compare, and choose the best strategy to follow regarding the integration. It is the time to take decisions, namely the ones related to the number of models we want to compare. We can compare two or more models simultaneously.

In the models comparison step the models are compared in order to determine correspondences between concepts and identify possible conflicts. Thus, it is necessary to identify inter-model relationships. This activity is known as correspondence between models, which is defined as follows: Given two models M1 and M2, for each concept identified in M1 we have to find another one in M2 that is semantically similar to the one identified in M1(Su).

According to Ehrig and Sure (2004), there is a set of rules that can be used to verify the similarity between IE of different models. The rules that we consider relevant to this work are:

- Rule 1 – If two IE have the same Uniform Resource Identifier (URI), they are equal.
- Rule 2 – If two IE have the same instances, they are equal.
- Rule 3 – If the description of two IE is similar, it is likely that the two IE are also similar.
- Rule 4 – If the attributes of two IE are equal, it is likely that the two IE are also equal.
- Rule 5 – If the name of two IE is the same or similar, it is likely that the two IE are equal or similar.
- Rule 6 – If the hierarchical path to the IE is equal, the compared IE are similar.
- Rule 7 – If the super-entities are the same, the compared IE are similar.
- Rule 8 – If the sub-entities are the same, the compared IE are similar.
- Rule 9 – If two IE have equal "sister IE", the compared IE are similar.
- Rule 10 – If two instances have the same "mother IE", they are similar.
- Rule 11 – IE that have equal instance quantity are similar.
- Rule 12 – If two instances are connected to another instance through the same property, they are similar amongst themselves.

In order to get to a correct correspondence we must follow a process which uses several of the rules above mentioned. In Ehrig and Sure (2004) a correspondence process between models is suggested, where the above identified similarity rules are used.

In the conforming models step the identified conflicts are solved, allowing the unification of the models. In order to achieve this, we can use several techniques such as type transformation, restructuring, renaming, etc. (Rizopoulos and McBrien 2009). In situations where it is not possible to solve the conflicts arising from basic inconsistencies, someone responsible for the models in question should guide the designer in the resolution of conflicts (Batini et al. 1986).

In the model unification step the unification of the models compared previously is made, resulting in a new model.

The most common transformations are described in Rizopoulos and McBrien (2009) and Rizopoulos (2009).

OBJECTIVES AND SOLUTION PROPOSAL

After having identified the problem and the research questions, as well as having analyzed the related work, we are now able to define the solution goals. The main objective of this work is:

- To propose a method for developing an IRA using a bottom-up approach that can assure an easy maintenance and a semantic interoperability

In addition, we also want to achieve the following objectives:

- To identify corresponding IE in different models
- To develop an IRA for the Portuguese Health Sector, with the most important IE for the Portuguese Health Sector

According to the above defined objectives, we now present the artifact that this research proposes to achieve those goals and to address the problem identified in this work.

In this work, the artifact consists of a method that uses a bottom-up approach to develop an IRA capable of ensuring an easy maintenance and semantic interoperability. This method is summarized in Figure 1.

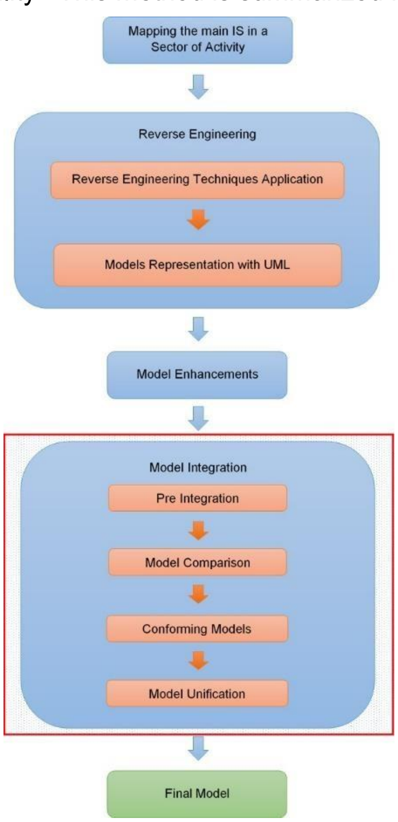


Figure 1: Solution Proposal Step

Next, we describe the different steps involved in the proposed method. It should be noted that the Model Integration step is the main focus in this research.

MAPPING THE MAIN IS IN A BUSINESS CATEGORY

In a given business category there can exist tens or hundreds of IS. In order to design an IRA, it's necessary to understand, with the people in charge for the IS of that particular business category, which are the most important IS that can form the basis for the creation of such IA.

With the identification of such IS, it becomes possible to identify the most important IE, attributes, and relationships for the sector, and thus initiate the development of the IRA.

REVERSE ENGINEERING

After the main IS of a particular business category have been identified and their database models analyzed, it is necessary to perform some transformations and proceed with their modeling. In this step there are two distinct activities:

1. Application of Reverse Engineering transformations
2. Model representation with UML® and XML

In the first activity, reverse engineering transformations are applied in order to make an abstraction of the analyzed models and tables and reach individual models for each of the analyzed IS. This abstraction allows us to get models where there is no reference to characteristic database information.

In the second activity the representation of models with UML and XML is carried out. The use of UML is related to the fact that this is the language chosen to represent the IA to develop. On the other hand, the use of XML allows us to represent the attributes of each IE separately, besides facilitating the model integration to be applied in the Solution Proposal of this work.

The final result of this phase is an individual model for each analyzed IS without characteristic database information.

MODEL ENHANCEMENTS

After the abstraction of models and database tables has been concluded, we proceed with their improvement. To achieve this, some model enhancement transformations are used, as well as UML good practices, such as IE generalization, IE specialization, inheritance, concept separation (Fowler 2003).

Here, we need to remember that there can be an IRA for a more general business category. In that case it is necessary and useful to consider the wider IRA, in order to improve interoperability between the sector under study and the broader sector.

The final result in this phase is an individual model for each of the analyzed IS, with enhancements that increase their understanding and correctness.

MODEL INTEGRATION

After enhancements in the models of the analyzed IS have been made, we proceed with their integration in order to finish the method proposed in this work and so get an IRA for a specific business category, where there is no duplication of corresponding entities.

Model Integration of the related work provides a more comprehensive and theoretical description regarding model integration. For this reason, it is necessary to frame the theme in the specific context of this work. To reach this purpose, the four steps were adjusted to the solution of model integration with special attention to the second phase – model comparison.

Pre Integration of the Solution Proposal

In the context of this work, and in order to achieve the objective of identifying IE corresponding to different models, only binary comparisons will be made; that is, only two models are compared at a time.

Model Comparison of the Solution Proposal

We will base ourselves in the process described in Ehrig and Sure (2004) to make the IE comparison of different models.

First, it is necessary to quantify the concept of correspondence in this work, set a minimum threshold for concluding that two IE are correspondent, and assign weights for each of the similarity rules described above. After that, the correspondence process to be used in this research is defined.

For the scope of this study, we consider that two names are correspondent if they have the same or synonymous labels, or if the name correspondence techniques allow us to establish which are the same or similar. Regarding the attribute correspondence, we consider that two IE are correspondent if at least 75% of the attributes of an IE have correspondence in the attributes of another IE. These parameters can be configured according to the needs and the degree of certainty required by the situation.

As far as the minimum threshold to conclude that two IE are correspondents (cut-off) is concerned, we choose the value of 0.75 because, following the process defined below in this section, this value is high enough to avoid false positives. This parameter can be configured according to the needs and the degree of certainty required by the situation.

Table 1 brings together the similarity rules described in the related work graduated by their level of importance. In Group 1 the most important rules to verify the similarity between IE are present and in Group 4 the less important ones. When drawing up the group, we considered that the model integration to carry out this work is based on a bottom-up approach, where the final model is achieved by applying bottom-up primitives, starting from basic concepts and building more complex concepts (Batini et al. 1992). For this reason, we consider that the attributes or the description of an IE (Rules 3 and 4) are more important than the name given to an IE (Rule 5).

Group 1	**Group 3**
Rules 1 and 2	Rule 5
Group 2	**Group 4**
Rules 3 and 4	Rules 6 to 12

Table 1: Grouping Similarity Rules

The assigned weight to each rule is given according to the equation:

$$Weight_{SimilarityRule} = \frac{1}{Group_{Number}}$$

Where $Group_{Number}$ refers to the group number that contains the similarity rule of interest.

Thus, Table 2 shows the weights of each of the similarity rules described in the related work of this document.

Rule	Weight	Rule	Weight
1	1	7	0.25
2	1	8	0.25
3	0.5	9	0.25
4	0.5	10	0.25
5	0.33(3)	11	0.25
6	0.25	12	0.25

Table 2: Assigned Weight by Similarity Rules

In this work, we will use the following process of model correspondence:

1. Given two models, we intend to calculate the similarities between any pair of EI.
2. Choose a pair of IE to be compared.
3. Iterate, in order, for all the similarity rule groups defined in the related work and apply no more than one rule of each group.
4. Sum weights of the rules applied.
5. When sum of weights is higher than the cut-off (0.75), IE are considered correspondent and the process ends.
6. If at the end of the process, the value obtained is lower than the cut-off (0.75), IE are not considered correspondent.

In case any evident correspondence is validated through this process, we should proceed to a contextual interpretation using, if necessary, someone responsible for the IS involved.

At the end of this step it will be possible to present the correspondent IE between different models.

Conforming Models of the Solution Proposal

By using the similarity rules and the correspondence process between models described above, conflicts are minimized. For example, we do not run the risk of two homonymous IE (same spelling and/or pronunciation but different meanings) be considered correspondent just because the descriptions and attributes of the two IE are taken into account.

Model Unification of the Solution Proposal

In this phase we proceed with the unification of the models of each analyzed IS. To accomplish this unification we use the transformations presented in Model Unification of the related work.

The final result of this step is a model that is the result of the unification of the analyzed IS, where there is no duplication of corresponding entities. This model is also the final output of the method suggested in this work to define an IRA for a particular business category.

DEMONSTRATION

We carry out a demonstration to clarify how the proposed solution allows us to solve the problem defined above, answer the research questions, and achieve the objectives defined.

In this demonstration, an instantiation of the proposed solution for the Portuguese Health Sector is implemented, culminating in the creation of an IRA for that sector.

In this document we will first make an instantiation with an academic example and afterwards we present the same method applied to the healthcare industry.

DEMONSTRATION WITH AN ACADEMIC EXAMPLE

In this demonstration we used an academic example where the steps of the solution proposal of this work are followed.

For a better understanding, this demonstration follows the structure:

1. Demonstration of the Reverse Engineering and Model Enhancements steps
2. Demonstration of the Integration Model step

Demonstration of the Reverse Engineering and Model Enhancement Steps

In this section we demonstrate the Reverse Engineering and Model Enhancement steps of the proposed solution of this work. The following academic example is considered.

Given five database tables (Figure 2), we obtain a model that conceptually represents these same tables and the relationships established between them.

On the table, *FN* indicates the number of the field of the table, *PK* indicates if a certain field is or is not Primary Key of the table, and *FK* indicates if a certain field is or is not Foreign Key of the table. The *Physician* table and the *Auxiliary* table represent concepts that are specifications of the *Person* concept.

Starting from the tables in Figure 2, it is necessary to understand how these tables are related to each other, which fields represent primary (PK) or foreign (FK) keys, which fields represent attributes that must be removed at a higher level of abstraction, etc.

Database table "Person"

FN	Field Name	PK	FK
1	personID	X	
2	name		
3	age		
4	genre		
5	addressID		X
6	dbCreatedBy		
7	dbSincDmlID		

Database table "Address"

FN	Field Name	PK	FK
1	addressID	X	
2	district		
3	city		
4	localitaly		
5	postalCode		
6	street		
7	apartment		
8	foreignAddress		
9	foreignLocality		
10	foreignPostalCode		

Database table "Physician"

FN	Field Name	PK	FK
1	physicianID	X	
2	name		
3	age		
4	orderNumber		
5	speciality		
6	workplace		
7	contactID		X
8	dbCreatedBy		

Database table "Contact"

FN	Field Name	PK	FK
1	contactID	X	
2	contactType		
3	number		
4	dbCreatedBy		

Database table "Auxiliary"

FN	Field Name	PK	FK
1	auxiliaryID	X	
2	workSchedule		
3	contactID		X
4	dbCreatedBy		

Figure 2: Academic Example's Database Tables

To do that, we followed the strategy described in the *Reverse Engineering* section of this article. Specifically, IE are developed (in the example, each table happens to be represented as IE), attributes that act as foreign keys are removed (e.g., attribute *addressId* of the *Person* table), attributes which only have meaning at the level of databases are also removed (for example, the attribute *dbCreatedBy* of the *Person* table). After all those modifications are identified and performed, UML is used to carry out the modeling, resulting in the model presented in Figure 3 (from now on referred as model A).

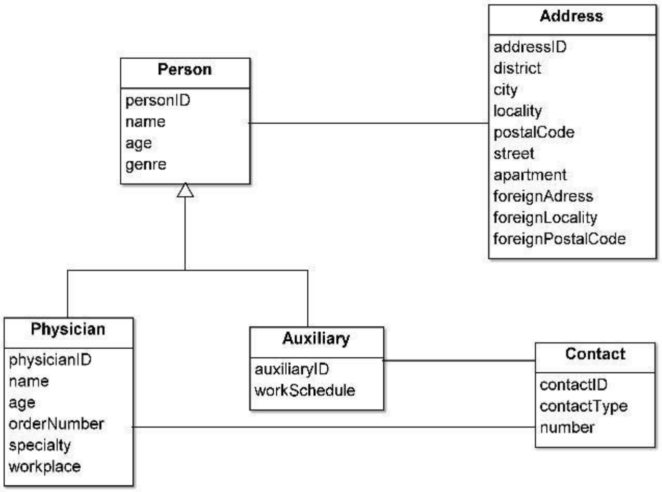


Figure 3: Academic Example's Model A after Reverse Engineering

Once we have reached a model where no data related to the database is present, it is necessary to enhance that model.

The strategy followed is described in the *Model Enhancement* of the related work. Summarizing it, the concepts separation is made (e.g., IE Address can be split into two IE, as it clearly represents the concept of Portuguese address and foreign address); redundant attributes are removed (attributes *Name* and *Age* in the *Physician* IE are redundant because they are also present in the *Person* IE); the relationship generalization is performed (the relations between the IE *Contact* and *Physician* and between the IE *Contact* and *Auxiliary* should be generalized, by removing these two relationships and creating a new relationship between *Contact* IE and *Person* IE).

After these changes are performed, we get the model presented in Figure 4.

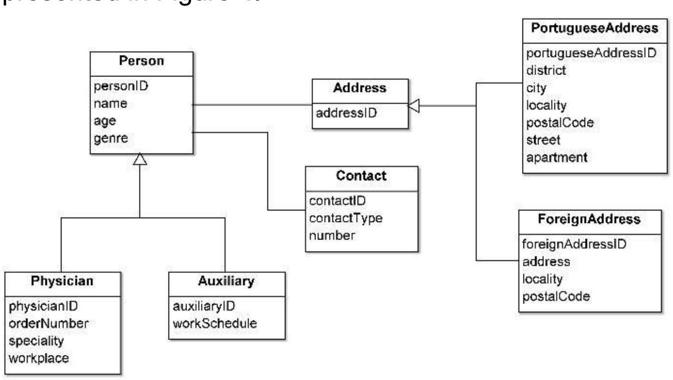


Figure 4: Academic Example's Model A after Model Enhancement

Demonstration of the Integration Model Step

In this section we intend to make the demonstration of the Integration Model step of the proposed solution of this work, where the phases of comparison and unification of models are specially treated. The following academic example is considered.

Given two models A and B, where model A refers to the outcome of the previous demonstration (Figure 4), and model B is shown in Figure 5, we initially intend to realize which IE are the correspondent IE in the two models. In this case, *A.Person* means: "IE of model A with the name *Person*".

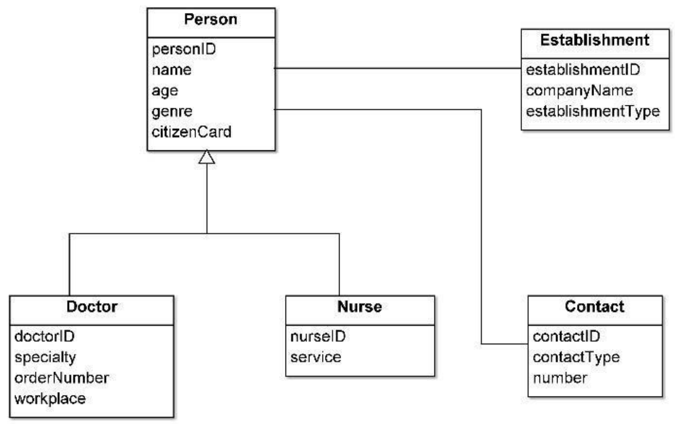


Figure 5: Academic Example's Model B

By simply analyzing the models in Figures 4 and 5, it is possible to estimate that the IE pairs *A.Person/B.Person*, *A.Physician/B.Doctor*, and *A.Contact/B.Contact* are correspondent. However, we must demonstrate that this idealization is true. For demonstration purposes, we will proceed by comparing only the second identified pair.

Iterating through the groups of the similarity rules defined in the related work on this document, the first rule that can be applied is Rule 4 (If the attributes of two IE are equal, it is likely that the two IE are also equal). To apply this rule, there must be a parity at the attribute level of 75% or more. We count the weight of this rule (0.5) in a variable starting at zero. Since the value accumulated in this variable does not exceed the cut-off established of 0.75, we must continue the iteration. So, the second rule to apply is the Rule 5 (If the name of two IE is the same or similar, it is likely that the two IE are equal or similar) because IE names are synonymous. By adding the weight of this rule (0.33) to the variable initialized earlier, we come to the value 0.88. Since this value already exceeds the cut-off of 0.75, we consider that the two IE are correspondent and the correspondence process comes to an end.

Following a similar reasoning, we conclude that the other two IE pairs previously mentioned also contain correspondent IE, and that there is no other pair of corresponding IE.

At the end of model comparison, we conclude that *A.Person* is correspondent to *B.Person*, *A.Physician* is correspondent to *B.Doctor*, and *A.Contact* is correspondent to *B.Contact*.

Based on these conclusions, we shall now proceed to solving the existing conflicts. In this example, there is a name conflict, since *A.Physician* and *B.Doctor* have synonym names. Therefore, we will rename one of these IE so that both keep the same name.

After solving all the conflicts, it is then possible to proceed with model unification. Exploring the unification techniques referred in the Model Unification of the related work, we will use the "*Merge IE*" transformation to collapse all the corresponding IE pairs.

The final outcome of this step is a model resulting from the unification of the analyzed IS, where there is no duplication of corresponding entities. Thus, we get the final unified model, which ensures that there is no duplication of corresponding IE. That model is presented on Figure 6.

Note that the resulting IE name of the union of the IEs *A.Physician* and *B.Doctor* is Doctor as it is the most usual term.

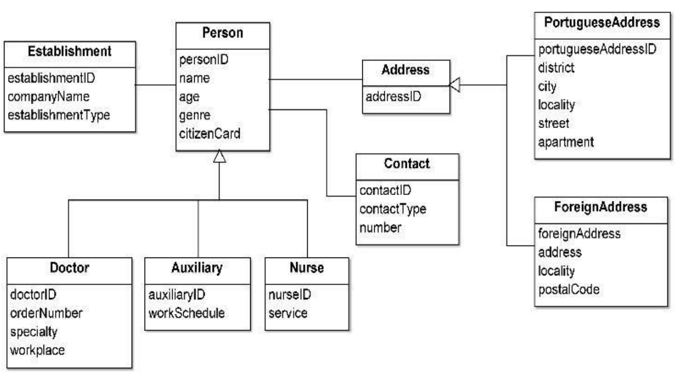


Figure 6: Academic Example's Model A and B Unification

INFORMATION REFERENCE ARCHITECTURE FOR THE PORTUGUESE HEALTH SECTOR

In this section we present the application of our proposal to the Portuguese Health Sector.

Although we applied the same logic of the previous demonstration, the instantiation described in this section does not intend to be so extensive, because it is impractical to describe every detail of it. Furthermore, for reasons of confidentiality, some of the documents and attributes may have been changed.

For a better understanding, this instantiation has the following structure:

1. Mapping the main IS of the Portuguese National Health Service
2. Reverse Engineering step
3. Model Enhancements step
4. Integration Models step

Mapping the Main IS of the Portuguese National Health Service

In this first step, we followed the strategy presented in the *Mapping the Main IS in a Business Category* section. Thus the mapping is made of the main IS of the Portuguese National Health Service (NHS). Together with the people in charge of the IS in the Portuguese Health Sector we came to the conclusion that the development of an IRA was of utmost importance. Concomitantly, it had to have as support the three IS considered as the pillars of the Portuguese NHS in the future. The three IS identified are the *Registo Nacional de Utentes* (RNU), *Registo Nacional de Profissionais* (RNP), and the *Sistema de Gestão de Entidades Informacionais* (SGES).

The final outcome of this step is the models and original tables of the aforementioned systems. For reasons of confidentiality, these models and tables are not presented in this document.

Reverse Engineering Step

Based on the models and original tables of the main Portuguese NHS (RNU, RNP, and SGES), we followed the strategy described in the *Reverse Engineering* section of the Solution Proposal and that can be easily understood in the demonstration with the academic example. This step allows us to reach a higher level of abstraction than the original, and ends with the UML modeling of the analyzed models. More particular to this instantiation it is possible to highlight the most relevant actions taken in this step. The concrete examples given for each of these actions are only intended to be explanatory and illustrative, not exhaustive.

- *IE Development* – tables will now be represented conceptually as EI.
- *Database model attribute removal* – attributes that are meaningful and useful only at the level of databases are removed. Particularizing to the analyzed IS, we found various attributes that fit this scenario. For example, in the initial models of the RNU and RNP we found attributes like *created_by*, *creation_date*, *last_updated_by*.
- *Foreign key removal* – attributes that function as foreign keys are removed from the IE and relationships are created between those IE and the IE where the attributes in question serve as primary

key. For example, in one of the early models analyzed, the *pro_profissid_personal_id* attribute is present in the ProProfissid IE as foreign key, and in the ProProfissidPersonal IE as primary key. That attribute is removed from the ProProfissid IE and a relationship is created between the two involved IE.

- *Database table removal* – some tables have meaning in the context of databases but at a higher level of abstraction, which aims to identify the main concepts of a particular model; these tables should be removed. For example, in the initial analyzed models we found tables such as IdutFamilyHist or ProTeamElemHist, which are nothing more than tables that keep records relating to historical information.

The final outcome of this step is an individual model for each analyzed IS without data related to the database modeled in UML. These models are shown in Figure 7 (RNU), Figure 8 (RNP), and Figure 9 (SGES).

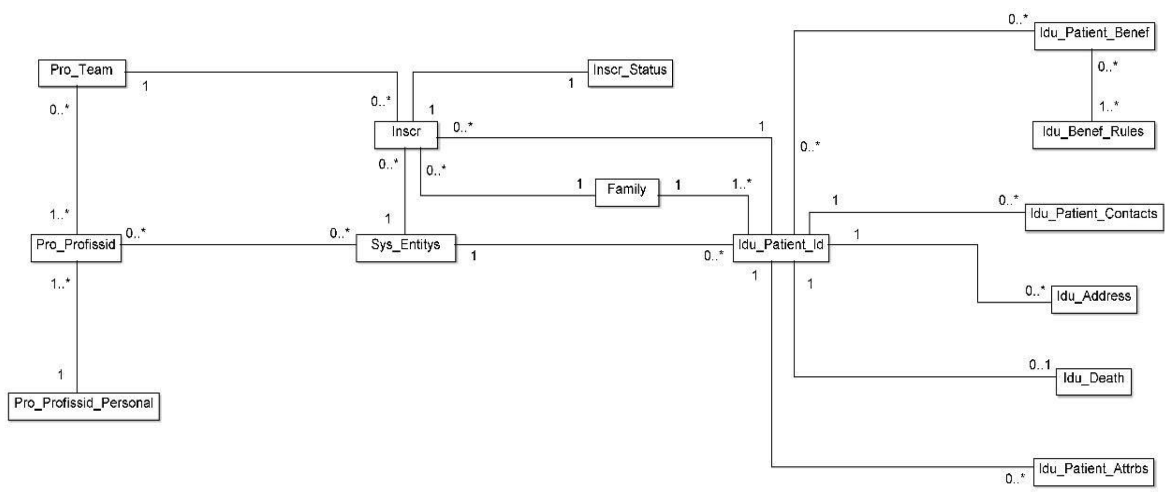


Figure 7: RNU Model after Reverse Engineering Step

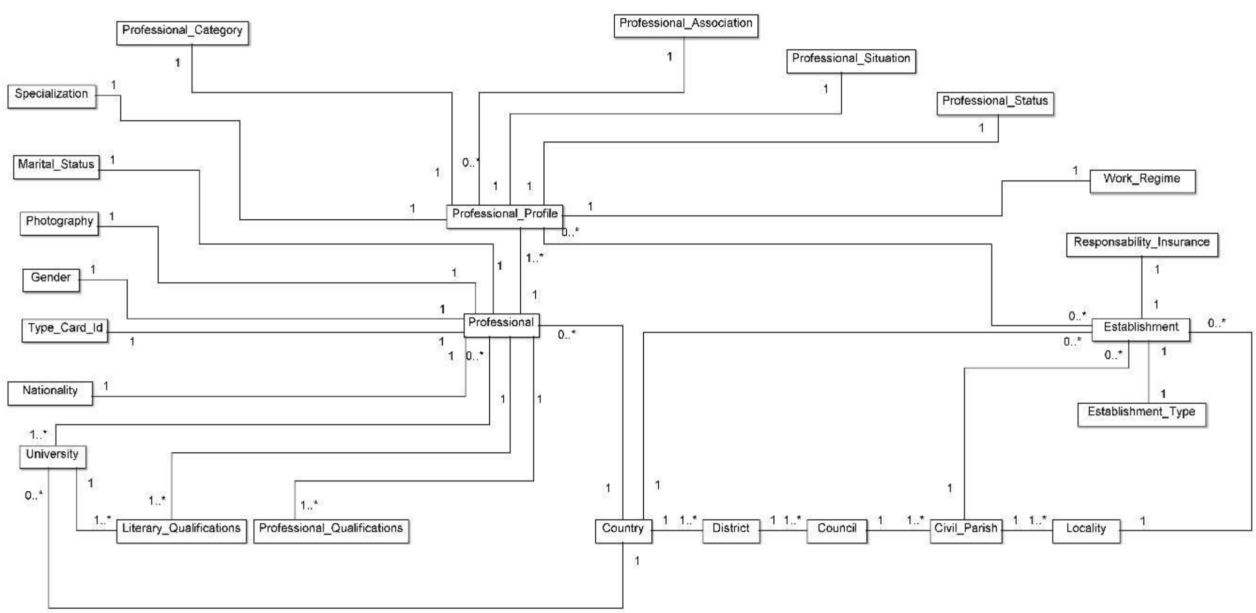


Figure 8: RNP Model after Reverse Engineering Step

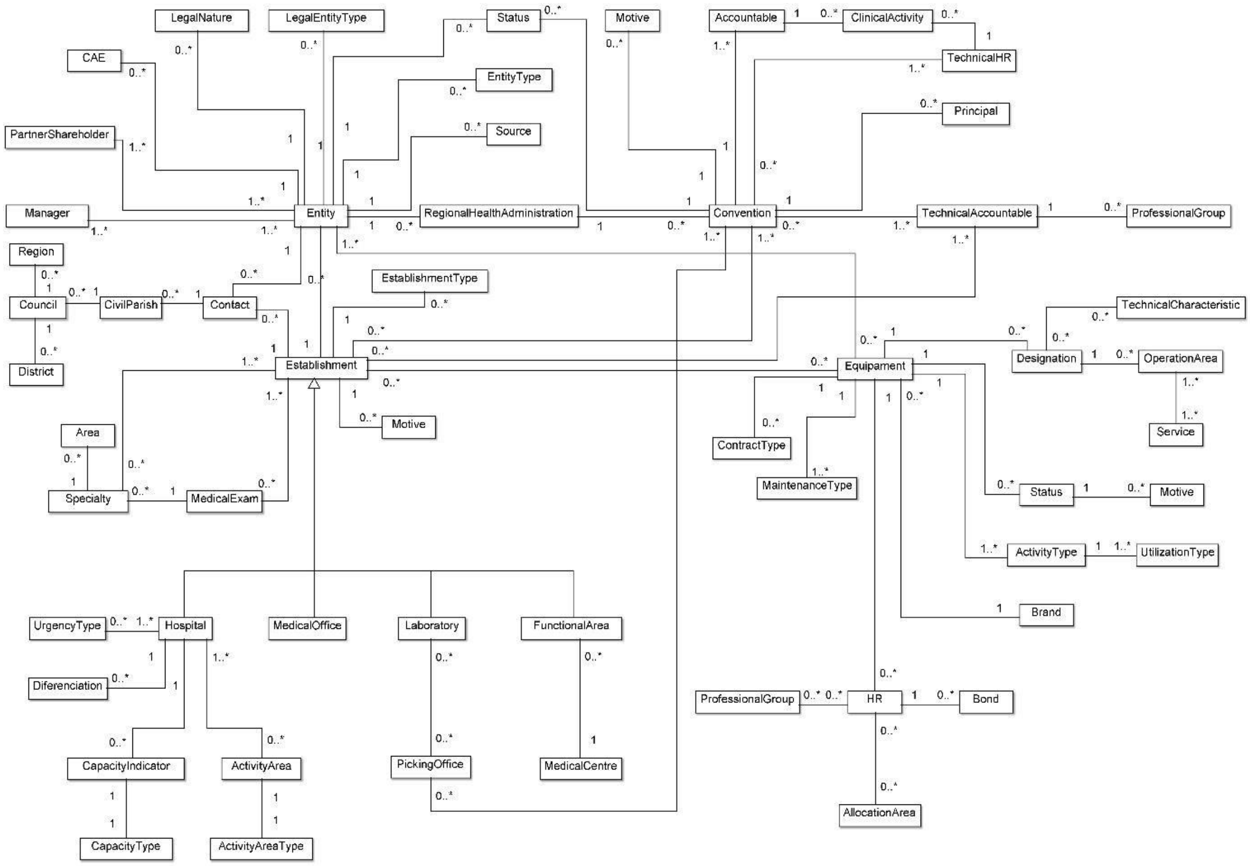


Figure 9: SGES Model after Reverse Engineering Step

Model Enhancements Step

Based on the individual models for each of the analyzed IS (RNU, SGES RNP), we followed the strategy described in the *Model Enhancements* section of the Solution Proposal and that can be easily understood in the demonstration with academic example. This step allows us to develop improved models with a higher correctness. More particular to this instantiation, it is possible to highlight the most relevant actions taken in this step. The concrete examples given for each of these actions are only intended to be explanatory and illustrative, not exhaustive.

- *IE Generalization* – the result of the union of two or more IE results in a new IE that keeps the common characteristics of the initial IEs. For example, in one of the analyzed models, the IE *Professional*, *PartnerShareholder,* and *Manager* can be generalized as a *Person* IE.
- *Attribute Generalization* – attributes that are repeated in two IE that share the same IE "mother" can be removed from their IE and placed in the IE "mother".
- *Concepts Separation* – for a better division of concepts, it is useful to carry out its separation. For example, in the analyzed models, the *Address* IE represents both the concept of Portuguese address as the foreign address. Thus, these concepts should be separated, resulting in the creation of a *PortugueseAddress* IE and a *ForeignAddress* IE.
- *Introduction of correct nomenclature* – for better model correctness, here we introduced some best practices related to nomenclature. For example, it was decided to standardize all the analyzed models, representing all IE names with the first letter capitalized, and all the attribute names with the first letter lowercase.

Since there is a common IRA for all the Portuguese PA (broader than the Portuguese Health Sector), on this instantiation we will use concept representations that are present in the Portuguese PA IRA. That is, in order to improve the alignment between the PA IRA and the analyzed IS, and because there is not a unique representation for some concepts (for example, the concepts Address, Contact, Birth), we have decided to represent these concepts just as they are represented in the Portuguese IRA.

The final outcome of this step is an individual final model for each analyzed IS with improvements to allow for better understanding and to correct it. These models are shown in Figure 10 (RNU), Figure 11 (RNP), and Figure 12 (SGES).

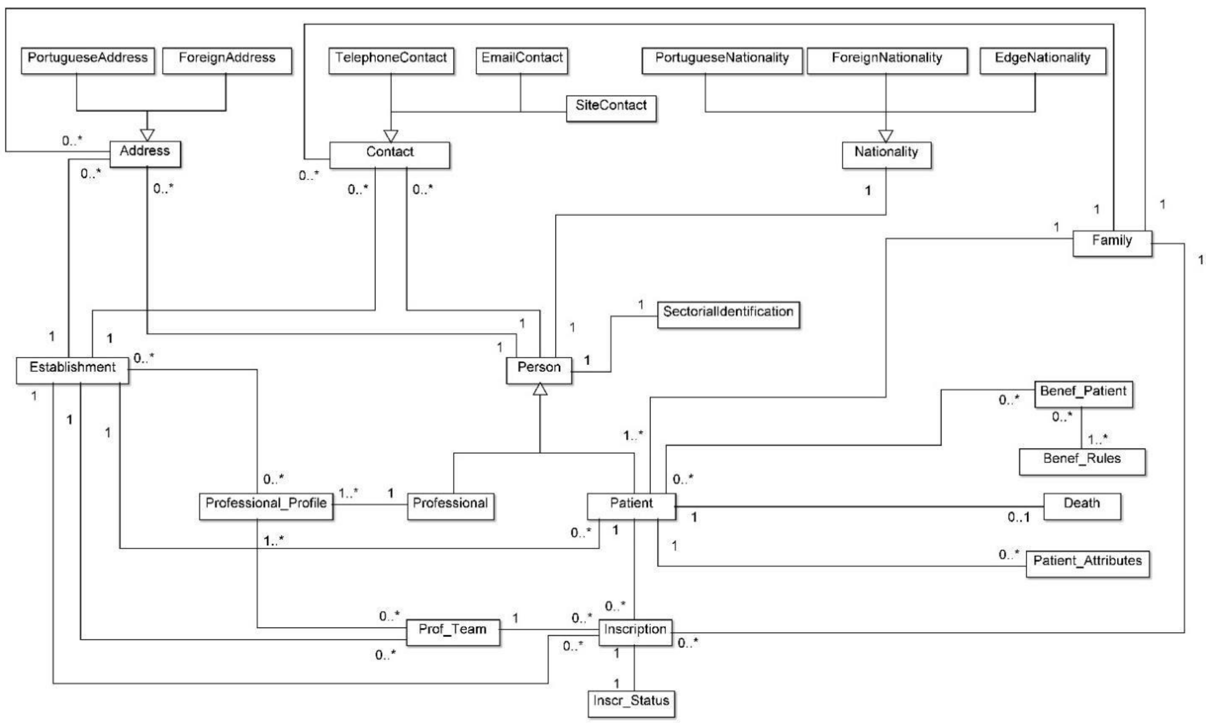


Figure 10: RNU Model after Model Enhancements Step

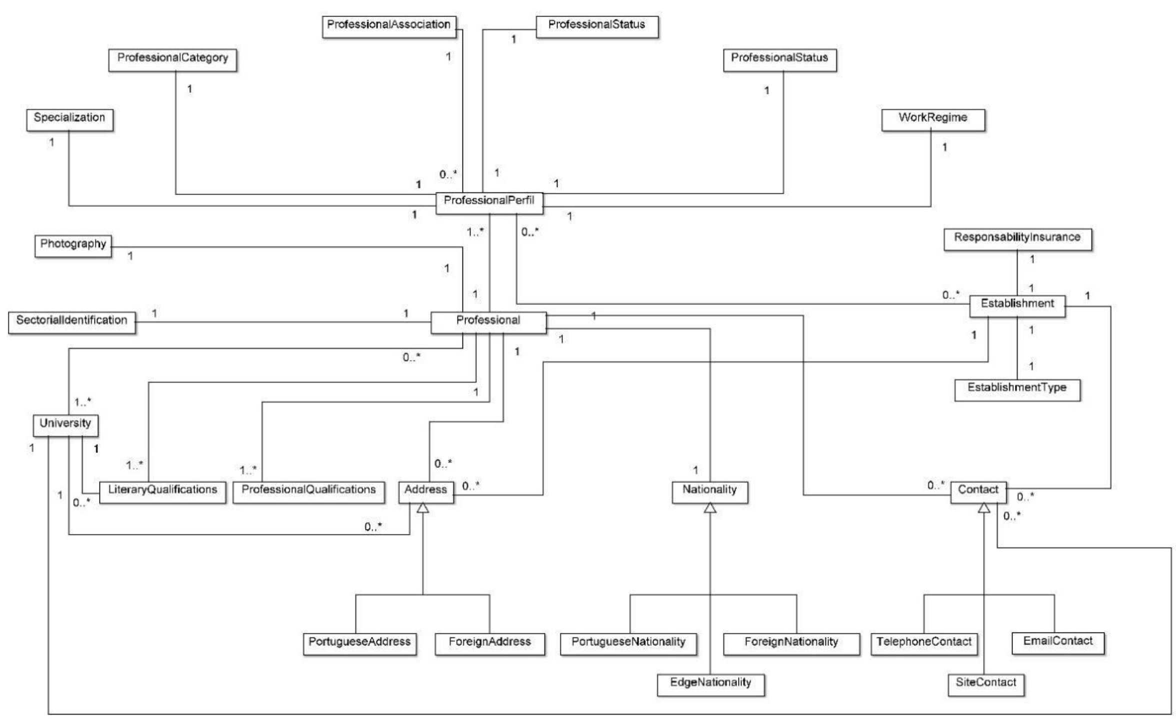


Figure 11: RNP Model after Model Enhancements Step

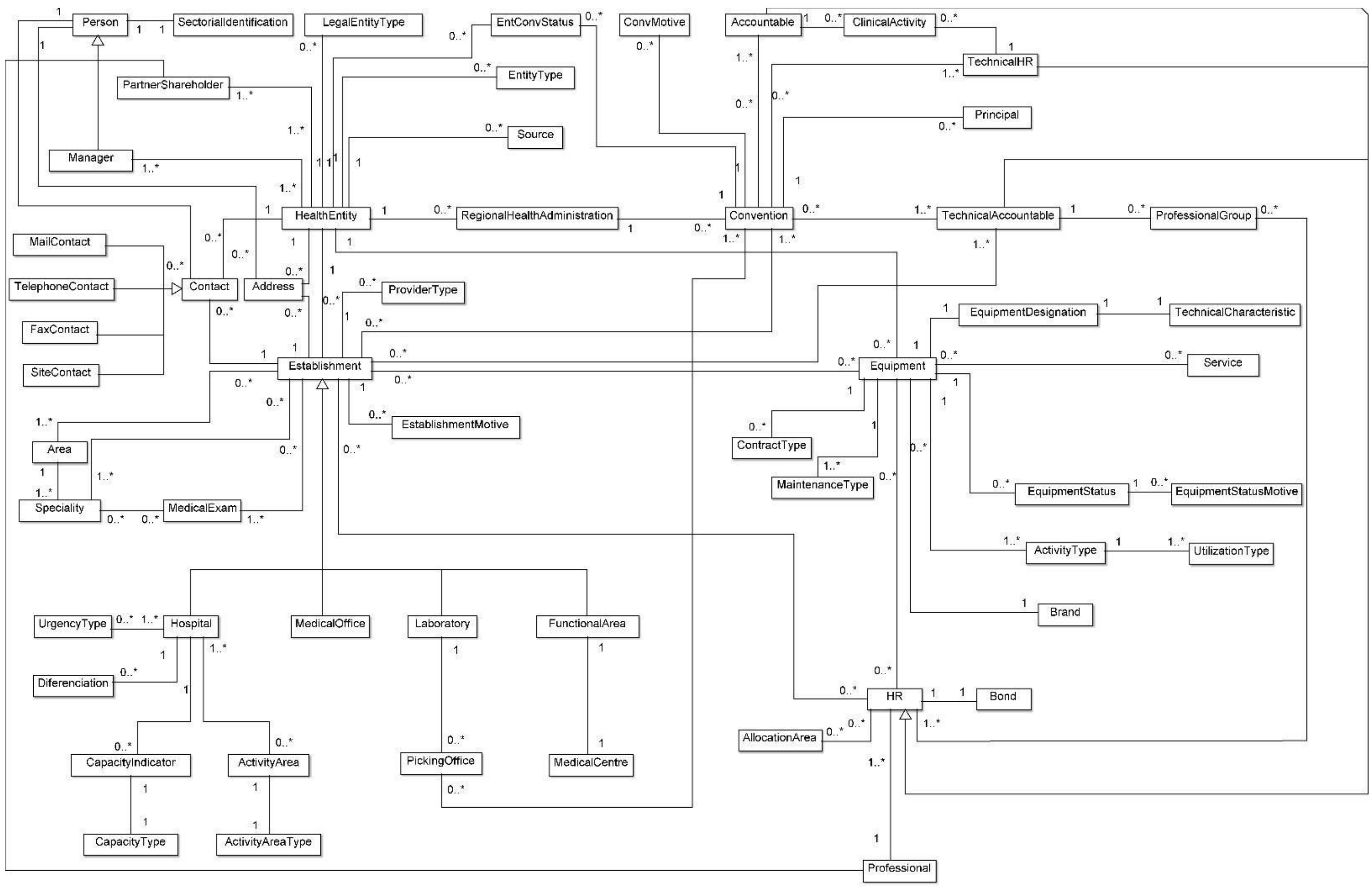


Figure 12: SGES Model after Model Enhancements Step

Integration Models Step

Based on the individual models for each of the analyzed IS (RNU, SGES RNP), we followed the strategy described in the *Integration Model* section of the Solution Proposal and that can be easily understood in the demonstration with academic example. More particular to this instantiation, it is possible to highlight the two major actions taken in this step. The concrete examples given for each of these actions are only intended to be explanatory and illustrative, not exhaustive.

- *Identification of corresponding IE between models* – after the analysis of the final individual models for each of the systems analyzed is used, the process of model correspondence of the solution proposal to identify all corresponding IE that can be collapsed is started. For example, IE representing concepts such as *Contact*, *Convention*, *Establishment*, *ProfissionalGroup*, *Address,* and *Professional* find a match on one or two of the analyzed models.
- *Junction of corresponding IE* – after all corresponding IE between the analyzed models is found, it is necessary to collapse all these IE to ensure that in the unified model there are no duplicated concepts. In this collapse, the resulting IE also keeps the union of all the relations of the two original IE. For example, when we join the *Contact* IE, which is present in the three analyzed models, we obtain in the final model a single *Contact* IE which contains all the relations of the original *Contact* IE.

The final outcome of this step is an individual model that is the result of the unification of the analyzed IS, where there is no duplication of corresponding entities. This model is also our proposal to IRA for the Portuguese Health Sector. This model is shown in Figure 13.

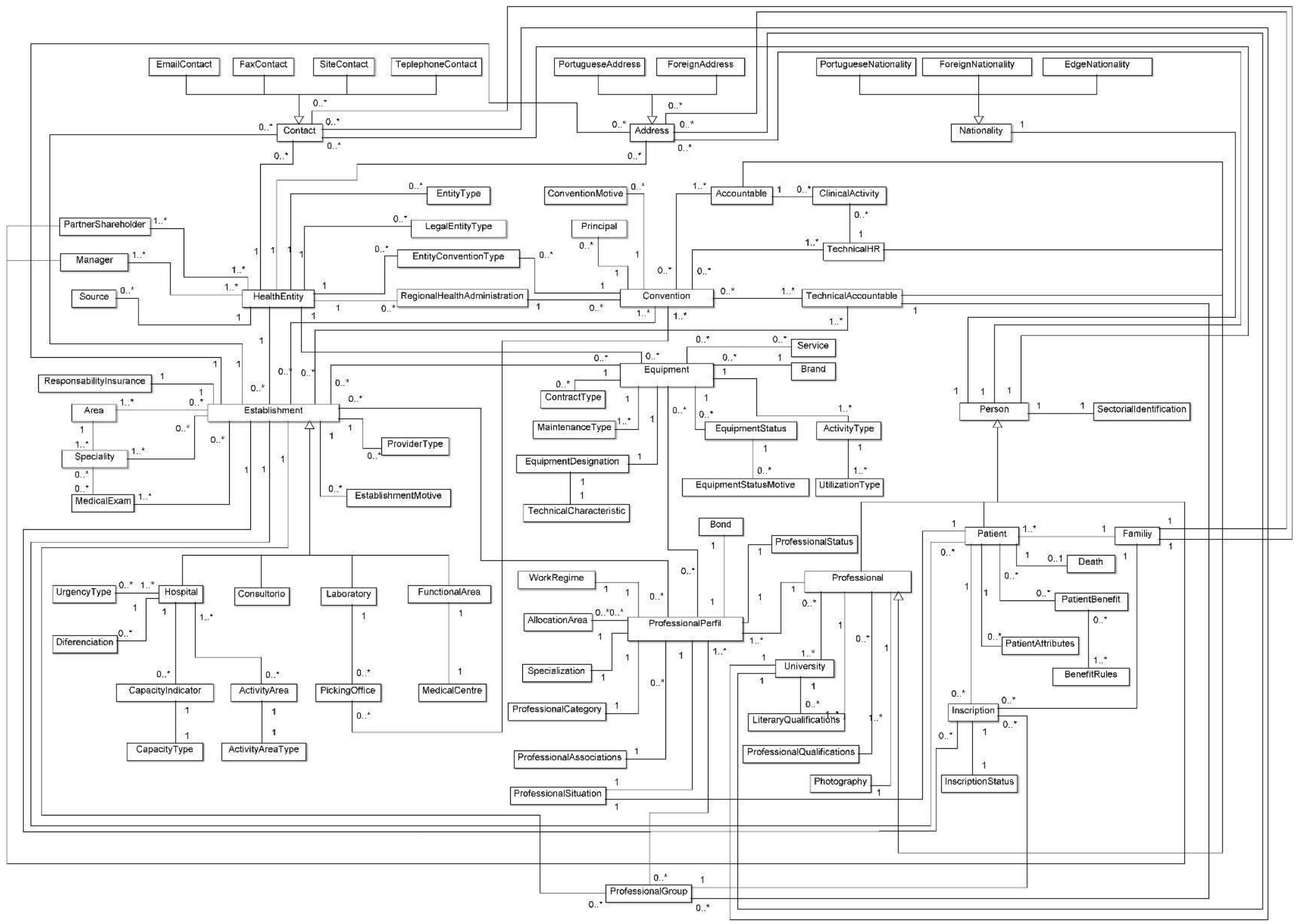


Figure 13: Proposal to IRA for the Portuguese Health Sector

EVALUATION

After having described the solution proposal and its instantiation, we present next the proposal evaluation.

Evaluation is considered one of the most important components in the DSRM since it is in this activity that we validate the contribution of the artifact developed to respond to the problem identified above as well as its utility, quality, and efficacy (Henver et al. 2004).

For better understanding the scope of the instantiation carried out to the Portuguese Health Sector, Figure 14 conceptually shows the current situation with respect to the motivation and the situation we want to achieve with this solution. The representation on the left illustrates the current motivation: there is no IRA for the Portuguese Health Sector to facilitate interoperability between the Portuguese Health Sector IS and the PA IRA; the oval shape represents the lack of an IRA that takes into account the Portuguese Health Sector IS. The representation on the right illustrates the solution proposed in this article: the creation of an IRA for the Portuguese Health Sector, which is a specification of the PA IRA and takes into account the Portuguese Health Sector IS and uses some of the best practices and PA IRA definitions and representations.

As stated in CIO (2010), using an IRA improves interoperability amongst models. Therefore, this solution leads, by itself, to the improvement of interoperability referred to previously.

The evaluation carried out in this work is done with the help of metrics adequate to evaluate, in a practical way, the proposed artifact – method for bottom-up development of an IRA that can ensure easy maintenance and semantic interoperability. To achieve this, we use the proposed model as an IRA for the Portuguese Health Sector, and compare it with the IS which forms the basis of its development, and with other works developed in a similar context.

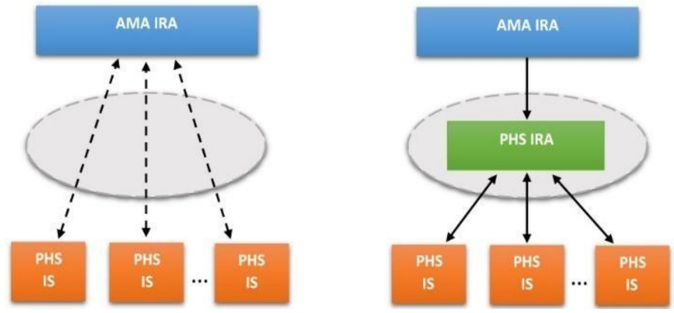


Figure 14: Current and Expected Situation

In these comparisons, the following metrics are used (Vasconcelos 2007; Vasconcelos et al. 2006):

- Number of IE
- Number of relationships between IE
- Number of attributes

These metrics are used to assess the dimension of an IA or of a model, and are directly related to the efficacy of their maintenance (Genero et al. 2003). That is, the lower the value of these metrics, the higher efficacy of maintenance of an IA or of a model.

Evaluation Conclusions

As mentioned at the beginning of this section, the fact that a specific business category adopts an IRA leads to an improvement in interoperability between models within that same category. Thus, we conclude that the instantiation of the work performed in the Portuguese Health Sector leads, in itself, to an increase in interoperability referred to above.

Furthermore, by implementing the metrics identified in this section it is possible to draw conclusions regarding the proposed method of bottom-up development of IRA that ensures ease of maintenance, and its instantiation for the Portuguese Health Sector.

As for comparisons with IS that underpinned the development of the model proposed in this work, it is possible to verify that the IRA for the Portuguese Health Sector contains less IE (Figure 15), fewer relationships between IE (Figure 16), and fewer attributes (Figure 17) than the sum of models that are the basis of the same IRA. Thus, it is possible to conclude that the proposed method in this investigation leads to IRA that are easier to maintain than the IS set that is at its basis.

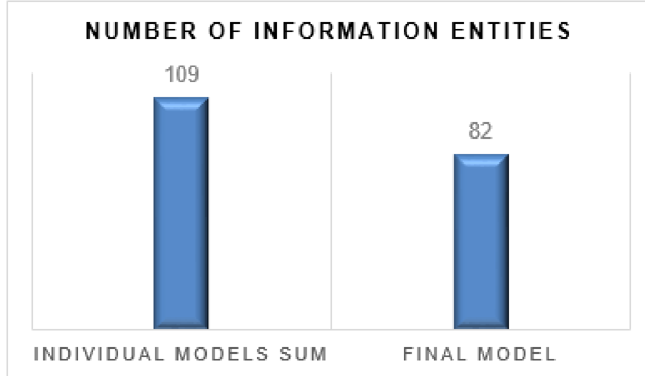


Figure 15: Comparison of the Number of IE between Individual Models Sum and Final Model

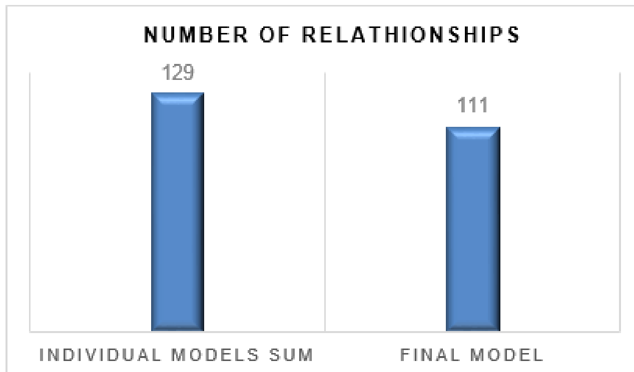


Figure 16: Comparison of the Number of Relationships between Individual Models Sum and Final Model

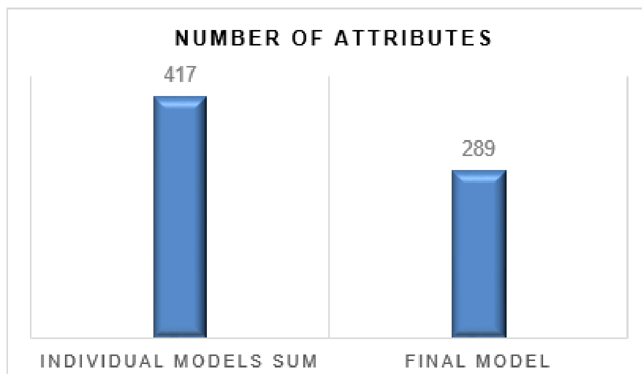


Figure 17: Comparison of the Number of Attributes of Individual Models Sum and Final Model

It is also possible to specialize this evaluation by referring to the conclusions drawn from the comparison of the suggested model as IRA for the Portuguese Health Sector with other similar works. The following sections describe the main conclusions from these comparisons.

Comparison with HL7 RIM

The fact that the proposed model in this work as IRA for the Portuguese Health Sector results from the integration of three different models, inevitably leads to the consequence of containing a larger number of IE, attributes, and relationships than the HL7 RIM. Therefore, it is possible to state that the IRA proposed has lower maintenance efficacy than the HL7 RIM.

On the other hand, the fact that HL7 is international, and doesn't take into account the specific context of the Portuguese Health Sector or the PA IRA, leads us to the conclusion that the proposed model ensures higher interoperability between the Portuguese Health Sector IS and PA IRA.

CONCLUSIONS

One of the identified ways to improve continuous competitiveness of organizations is through the creation of IA that can help in the sharing and exchanging of information or in the fast and easy development of IS.

As far as business categories are concerned, the lack of an IRA to guide and restrict the instantiations of a set of architectures and individual solutions leads to interoperability problems, translating, among other things, great difficulties in maintaining the existing IS.

The solution that this analysis proposes is defined as a method that uses a bottom-up approach and comes from a set of IS to get to an IRA that will be easy to maintain. To do so we used reverse engineering, model enhancements, and model integration techniques.

The method suggested in this study and its instantiation for the Portuguese Health Sector were assessed with the help of evaluation metrics that allowed us to reach conclusions concerning the maintenance of an IA or model.

Main Contributions

We believe that the proposed solution described in this work adds some value to the context in which it is inserted: the creation of IRA to ensure ease of maintenance and semantic interoperability.

The objectives defined for this research were successfully achieved, and each one of them represents an important contribution of this work.

As general contributions we note the creation of a method for the bottom-up development of an IRA that can ensure both ease of maintenance and semantic interoperability, and the identification of IE correspondence between different models. This method uses reverse engineering, model enhancements, and model integration techniques.

As far as the specific contributions for the Portuguese Health Sector are concerned, we emphasize the identification of the most relevant IE, as well as the development of an IRA.

Limitations and Future Work

In spite of the contributions coming from this solution proposal, there are some limitations that must be considered. These limitations can be directly associated to improvements to be worked in the future.

Thus, we suggest a deeper investigation on the Reverse Engineering and Model Enhancement steps of the suggested model; the instantiation of the suggested model to other business categories; the development of other evaluation metrics to assess the international level of an IA; and the use of HL7 RIM to improve the suggested model of this work for an IRA in the Portuguese Health Sector.

ABOUT THE AUTHORS

André Vasconcelos (andre.vasconcelos@ist.utl.pt) is Assistant Professor at Instituto Superior Técnico, Portugal.

Tiago Bras (tiago.g.bras@ist.utl.pt) holds an MSc in Information Systems and Computer Engineering at Instituto Superior Técnico, Portugal.

REFERENCES

Agência para a Modernização Administrativa:Arquitetura Informacional; refer to: https://m6.ama.pt/docs/ArquiteturaInformacional.pdf (accessed 4-10-2015).

S. Bagui, R. Earp: Database Design Using EntityRelationship Diagrams, Auerbach Publications (2003).

C. Batini, et al: A Comparative Analysis of Methodologies for Database Schema Integration, ACM Computing Surveys 18(4) (1986).

C. Batini, et al: Conceptual Database Design: An EntityRelationship Approach, The Benjamim/Cummings Publishing Company, Inc. (1992).

A. Bernstein: Middleware: A Model for Distributed System Services, ACM (1996).

E.J. Chikofsky, J.H. Cross: Reverse Engineering and Design Recovery: A Taxonomy, IEEE 7(1) pp.13-17 (1990).

L. Chiticariu: Enabling Data Interoperability with Provenance and Schema Integration, University of California (2008).

CIO, Office of the DoD: Reference Architecture Description (2010).

M. Ehrig, Y. Sure: Ontology Mapping – An Integrated Approach, LNCS (2004).

M. Fowler: UML Distilled: A Brief Guide to the Standard Object Modeling Language, Pearson Education (2003).

M. Genero, et al: Defining and Validating Metrics for Assessing the Maintainability of Entity-Relationship Diagrams (2003).

M. Godinez, et al: The Art of Enterprise Information Architecture: A Systems-Based Approach for Unlocking Business Insight, IBM Press (2010).

Health Level Seven International; refer to: www.hl7.org/implement/standards/rim.cfm (accessed 16-09-2014).

A.R. Henver, et al: Design Science in Information Systems Research, MIS Quarterly 28(1) pp.75105 (2004).

Interoperability Solutions for European Public Administrations: European Interoperability Framework (EIF) for European public Services (2010).

M. Lankhorst, et al: Enterprise Architecture at Work, Springer (2005).

E. Rahm, P. Bernstein: A Survey of Approaches to Automatic Schema Matching, The VLDB Journal 10(4) (2001).

N. Rizopoulos, P. McBrien: Schema Merging Based on Semantic Mappings, Springer-Verlag (2009).

N. Rizopoulos: Schema Matching and Schema Merging Based on Uncertain Semantic Mappings, University of London (2009).

S.H. Spewak, S. Hill: Enterprise Architecture Planning: Developing a Blueprint for Data, Applications, and Technology, QED Technical Publishing Group (1993).

X. Su: A Text Categorization Perspective for Ontology Mapping, Department of Computer and Information Science, Norwegian University of Science and Technology.

H. Tran. et al: View-Based Reverse Engineering Approach for Enhancing Model Interoperability and Reusability in Process-Driven SOAs, High Confidence Software Reuse in Large Systems, Springer Berlin Heidelberg, pp.233-244 (2008).

A. Vasconcelos, T. Brás: Information Reference Architecture for the Portuguese Health Sector, IEEE Conference on Business Informatics (2015).

A. Vasconcelos, et al: Information System Architecture Metrics: An Enterprise Engineering Evaluation Approach (2006).

A. Vasconcelos: Arquitecturas de Sistemas de Informação no Contexto do Negócio, Dissertação para a Obtenção do Grau de Mestre em Engenharia Informática e de Computadores (2001).

A. Vasconcelos: Arquitecturas dos Sistemas de Informação: Representação e Avaliação. Dissertação para a Obtenção do Grau de Doutor em Engenharia Informática e de Computadores, Instituto Superior Técnico, Universidade Técnica de Lisboa (2007).

R.W. Watson: An Enterprise Information Architecture: A Case Study for Decentralized Organizations, IEEE Computer Society (2000).


TOGAF®9 Training
15% Discount
To All AEA Members!
ENTERPRISES
You know you need quality training in Enterprise Architecture standards. But with so many training providers who do you go to?
Don't chance it - speak to a globally recognised company who have trained more people in TOGAF than any other.
Contact us today to discuss your requirements.
Australia: +61 2 8860 6419
Asia: +61 2 8860 6419
EMEA: +44 1400 567 567
USA: +1 203 945 8758
Or e-mail us at: hello@ate.enterprises
THE Open GROUP
TOGAF® 9
Accredited Training Course

Article

Enterprise Architecture Practice in Retail: Problems and Solutions

Svyatoslav Kotusev, Mohini Singh, and Ian Storey

Abstract

Currently Enterprise Architecture (EA) is widely practiced in different organizations working in diverse industries across the globe. Although it is generally acknowledged that there are no universal one-size-fits-all approaches to EA practice suitable to all organizations and industries, features and peculiarities of the approaches to EA followed in different industries are still poorly understood. In this article I analyze the EA practice in a large Australian retail chain operating in the fast-moving consumer goods business, discuss the industry-specific challenges with EA experienced by this company, and describe their potential solutions and mitigation strategies followed by the company.

Keywords

Enterprise Architecture (EA), Retail, Problems, Solutions

INTRODUCTION

An Enterprise Architecture (EA) can be represented by a collection of documents describing an enterprise from an integrated business and IT perspective intended to bridge the communication gap between business and IT stakeholders and, thereby, to improve business and IT alignment. An EA practice implies the development and use of a variety of documents typically called EA artifacts that translate global executive-level strategic decisions into specific information systems supporting them (Ahlemann et al. 2012).

Currently, EA is widely practiced in different organizations across the globe working in diverse industries, including banking (Gerber et al. 2007; Gonzalez 2011; Murer et al. 2011), agriculture (Hungerford 2007; Hungerford 2009), healthcare (Venkatesh et al. 2007), academia (Anderson et al. 2009), as well as a multitude of other sectors (Lynch 2006; Pheng & Boon 2007; Rees 2011; Richardson et al. 1990; Smith et al. 2012). At the same time, it is generally acknowledged that there are no universal one-size-fits-all approaches to EA practice suitable to all organizations and industries (Kotusev et al. 2015; Park et al. 2013; Saha 2009). However, despite the evident diversity of the industries where EA is successfully practiced, features and peculiarities of the approaches to EA followed in different industries are still poorly understood. Essentially, there is little information available on the specificity of an EA practice in different industries. On the one hand, it is not clear which characteristics of an organization's industry are significant and should be taken into account from the perspective of an EA practice. On the other hand, it is not clear exactly how these characteristics of an organization's industry may influence the design of an EA practice in such an organization.

In order to better understand the influence of a specific industry on an EA practice I studied in detail the EA practice in a large and widely known Australian retail chain (that wished to remain anonymous) operating largely in the fast-moving consumer goods business and then analyzed industry-specific features of this EA practice. The retail industry, and especially its fast-moving consumer goods segment, is characterized by high sales volumes, low margins, fast stock turnover, and heavy reliance on complex logistic networks for goods delivery and storage. From the management perspective the competitive position of a retail chain largely stands upon three pillars: lowering the cost, increasing the revenue, and improving the customer experience. The Australian retail market is very dynamic, highly competitive, and influenced by aggressive new market entrants. Companies are constantly competing on price and struggling to increase their market shares, while continually accommodating changing legislation. Moreover, companies have to respond quickly to their competitors' moves in order to stay afloat. Therefore, the fast-moving consumer goods retail business in Australia is very fast-paced, cost-sensitive, and reactive. Its business environment is highly competitive, rapidly changing, and largely unpredictable.

This article continues as follows: (1) I provide a brief overview of the studied organization, (2) I describe the "nuts and bolts" of the EA practice in this organization, and (3) I discuss the influence of industry on the EA practice including the most pressing industry-specific problems and their potential solutions.

COMPANY OVERVIEW

FMCG (fictitious name to ensure anonymity) is a major player in the fast-moving consumer goods retail market in Australia. It has multi-billion dollar revenues and employs tens of thousands of people, including several hundred IT staff and a similar number of its partners' outsourced IT personnel. The company is split into several lines of business and operates several hundred retail outlets across Australia.

FMCG largely implements the Diversification operating model (Ross 2005; Ross et al. 2006; Weill & Ross 2009) since different lines of business within FMCG are relatively independent, do not follow globally standardized business processes, and do not share global data, except for the supporting functions; for instance, finance and HR. However, each line of business implements the Unification operating model (Ross 2005; Ross et al. 2006; Weill & Ross 2009) because all retail stores within each line have standardized processes, IT systems, and shared databases for their main data types; for instance, logistics and products. Each line of business has its own IT delivery function.

Initially the strategic architecture for FMCG was planned by solution architects on an unsystematic basis. The first attempts to start practicing true EA can be dated back to 2007 when the role of enterprise architect was established. However, due to the fast-paced and reactive nature of FMCG's business, its architecture function has since undergone a number of transformations. The meaning of the enterprise architects' role at FMCG has been periodically redefined and the number of enterprise architects at FMCG has changed accordingly.

For a period of time FMCG had a very strong and influential centralized EA team which tried to proactively plan the strategic architecture for the whole organization, but this team was considered too bureaucratic by senior business stakeholders since all IT investments needed their agreement and approval. As described by an enterprise architect:

"Architects told the business what the business was gonna do and because of that the business felt they weren't listening and getting what is required to respond. And [business] will go and get frustrated."

At one point FMCG had one enterprise architect for each line of business, largely responsible for managing solution architects working for that line, but this model did not work well either because enterprise architects concerned with people leadership were unable to also produce a meaningful proactive strategic vision for the organization.

Currently, FMCG has a centralized and lean EA team focused more on identifying new IT-enabled business opportunities for growth and operational efficiency, than on ensuring compliance and governance. However, FMCG is still in the process of refining its own company-specific way to practice EA.

EA PRACTICE AT FMCG

In this section I will describe FCMG's architecture function, documents, and processes.

Architecture Function

Currently, FMCG has a centralized architecture function for the whole organization that includes enterprise and solution architects and is managed by the head of architecture, who reports directly to the CIO. The EA team is responsible for company-wide strategic architecture planning and consists of two enterprise architects reporting to the head of architecture. The solution architecture team is responsible for project-level architecture planning and consists of 12 solution architects reporting to the manager of architecture, who also reports to the head of architecture. Additionally, apart from the central architecture function, IT delivery functions of different lines of business have independent teams of application architects, domain and subject matter experts responsible for detailed technical designs of ongoing IT projects. The structure of the architecture function at FMCG is shown in Figure 1.

Architecture Documents

The EA practice at FMCG is based on 12 distinct types of documents produced by architects with the necessary involvement of other relevant stakeholders. Architecture documents used at FMCG with their brief description, meaning, developers, users, and purpose are described in Table 1.

FMCG does not use any specific software tools for developing, storing, and managing architecture documents. All documents are developed with the standard Microsoft® Office suite (PowerPoint®, Word, and Visio®) and stored in the central SharePoint® repository with the exception of inventories that were initially stored as Excel® spreadsheets, but eventually migrated into a ServiceNow™ Configuration Management Database (CMDB).

FMCG also does not have any standards on the use of different architecture modeling languages. The ArchiMate® modeling language is occasionally used by some architects for high-level technical diagrams and UML® is occasionally used for more detailed drawings, while the majority of architecture documents do not adhere to any specific modeling notations.

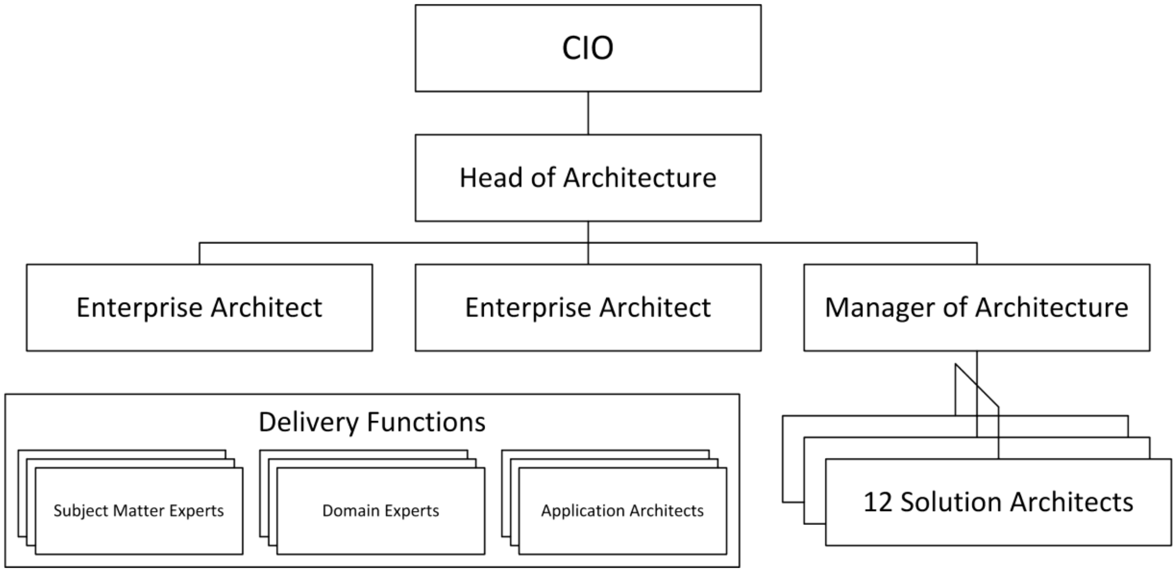


Figure 1: Structure of the Architecture Function

Table 1: List of Architecture Documents

Documents	Description
Strategic Papers	Strategic papers are very high-level analytical documents discussing the potential influence and impact of disruptive technical trends on the company's business. Essentially, they represent the results of a SWOT (strengths, weaknesses, opportunities, and threats) analysis from the technology perspective. Strategic papers are produced collaboratively by enterprise architects and senior business managers and communicated to a wide circle of business and IT stakeholders to inform their decision-making.
Principles	Principles are abstract global architecture maxims relevant for all IT solutions in the organization. Principles range from common IT policies found in many organizations, such as "reuse before buy, buy before build", to highly company-specific policies, such as "all store solutions should be robust to intermittent connectivity and network failure". Principles are formulated by enterprise architects and approved by senior business stakeholders. All project-level architectures developed by solution architects should be aligned with principles.
Business Capability Model	The business capability model is a one-page diagram describing business capabilities of the whole organization up to two or three nested levels of abstraction. The business capability model is maintained by enterprise architects and used primarily to facilitate a conversation with business stakeholders and prioritize IT investments. However, it is also used by solution architects and project managers for identifying the stakeholders, impact, and potential disruption of a solution.
Business Reference Architectures	The business reference architectures describes the desired ideal organization of business processes according to recognized industry best practices in certain important business capabilities. Business reference architectures are developed collaboratively by business stakeholders and enterprise architects and then used for identifying best opportunities for improvement and IT investments.
Roadmaps	Roadmaps are business-focused documents describing desired future IT investments and their impact in certain important areas for three years ahead. Roadmaps are written in business language and aimed at answering core questions of relevant stakeholders. Roadmaps describe planned IT investments through different "lenses", including financial, value, capability, structure, and other lenses. Roadmaps are developed collaboratively by enterprise architects and business stakeholders and used for making decisions on future IT investments and prioritizing them.

Documents	Description
Technical Reference Architectures	Technical reference architectures are high-level descriptions of the current and ideal target states of the IT landscapes supporting certain business capabilities. They are purely technical and IT-specific in nature. Technical reference architectures exist for 60-70% of business capabilities, but only 20-30% of business capabilities have their ideal future states described. They are developed by enterprise architects and used by solution architects to facilitate detailed project planning by providing a description of the current state as well as a description of the desired state that their projects should be aiming to achieve.
Inventories	Application, infrastructure, and information inventories are catalogs of the corresponding entities available in the organization. Application and infrastructure inventories are fully populated, while an information inventory list is still incomplete. Inventories are maintained collaboratively by enterprise and solution architects and used mostly as reference materials by solution architects to facilitate the project-level architecture planning.
Standards	Standards are specific technical recommendations relevant for all IT solutions in the organization; for instance, that all solutions should be based on the Microsoft .NET platform or that all customer-facing mobile apps should support both iOS® and Android™ platforms with native applications. Standards are set collaboratively by enterprise and solution architects, typically in a bottom-up manner as a result of a particular project introducing a new technology or specific need. All project-level architectures developed by solution architects should be compliant with standards.
Solution Overviews	Solution overviews are high-level documents describing specific IT solutions. The level of detail in solution overviews is abstract enough to be understandable for business stakeholders, but is specific enough for obtaining approximate estimates of time, cost, and risk. Solution overviews are developed for each project by solution architects with an input from domain and subject matter experts. Solution overviews are typically used by business stakeholders and architects for initial project discussions and approvals. Solution overviews also provide estimates for informing formal project business cases and serve as a basis for detailed project architectures.
Business Cases (not architecture documents, but important for the EA practice at FMCG)	Business cases are formal financial documents for specific IT solutions. Business cases specify anticipated quantitative measurable benefits, costs, and return on investment (ROI) for particular projects. Business cases are prepared collaboratively by business stakeholders and solution architects based on the estimates derived from solution overviews of the corresponding projects. Business cases are the main project-level documents used by business stakeholders to approve all IT solutions.
Solution Architecture Documents (SADs)	Solution Architecture Documents (SADs) are detailed technical descriptions of specific IT solutions. SADs are developed for each project by solution architects with an input from domain and subject matter experts after the solution overview and business case for this project have been approved. SADs are used predominantly by project teams, including application architects, domain and subject matter experts, for producing detailed designs and delivering projects.
Key Design Decisions (KDDs)	Key Design Decisions (KDDs) are summary documents describing significant architectural decisions taken for specific IT solutions, the reasoning behind them, their justifications, and pros and cons. For instance, KDDs should explain any deviations of a solution from established principles, standards, roadmaps, or technical reference architectures. KDDs are extracted from solution overviews and SADs by solution architects and used by enterprise architects and business stakeholders as main points of discussion, review, and approval for all IT solutions.

Architecture Processes

Architecture processes constituting the EA practice at FMCG can be roughly separated into enterprise-level processes and project-level processes. Enterprise architects are the main actors of enterprise-level processes, while project-level processes are carried out largely by solution architects.

Enterprise-Level Processes

Enterprise-level architecture processes at FMCG are mostly unstructured and not formalized. They consist of eight distinct activities of enterprise architects. These activities are largely independent of each other and can be carried out in parallel without any particular predefined order. Therefore, they are discussed starting from more "generic" activities and ending with more "specific" ones:

1. Enterprise architects monitor relevant technology trends in the external environment, communicate with senior business stakeholders, and periodically produce strategic papers with the analysis of the possible impact and influence of these trends on the organization.
2. Enterprise architects formulate architecture principles for the whole organization and discuss them with senior business stakeholders.
3. Enterprise architects maintain the business capability model and use it for discussions with senior business stakeholders in order to understand in which capabilities the IT investments should go.
4. Enterprise architects together with senior business stakeholders develop business reference architectures for important business capabilities by means of adapting established industry best practices to the FMCG's environment.
5. For the most important business capabilities enterprise architects develop IT investment roadmaps agreed with the relevant business stakeholders.
6. Enterprise architects develop and maintain technical reference architectures for important business capabilities according to their best understanding of the business needs and direction.
7. Enterprise architects maintain the technical inventories to adequately reflect the currently available IT assets.
8. Enterprise architects together with solution architects maintain and update enterprise-wide technical standards for IT project implementation.

Enterprise-level architecture processes at FMCG are shown in Figure 2.

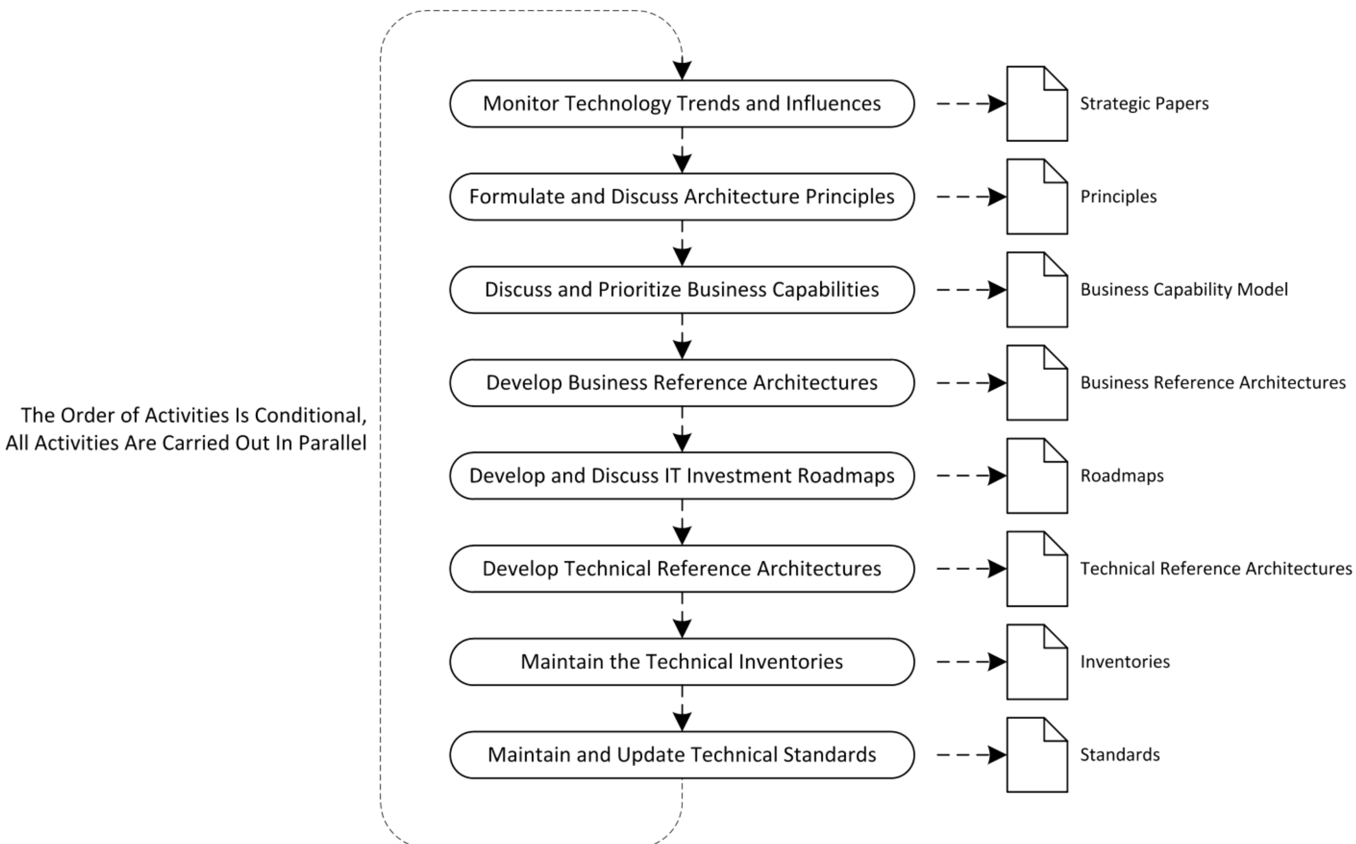


Figure 2: Enterprise-Level Architecture Processes

Project-Level Processes

Project-level architecture processes at FMCG are well-structured and largely revolve around two distinct governance bodies: the Innovation Forum and the Architecture Review Forum (ARF). The Innovation Forum is a governance body for testing and approving ideas for projects. It meets every two weeks and engages senior business and IT leaders, including finance officers responsible for the budgeting process. All IT projects are presented at the Innovation Forum where business and IT leaders evaluate the viability of each project from the business perspective based on its estimated cost, value, benefits, maintainability, risk, and other factors. Only worthwhile projects are given approval and necessary funding. However, the most significant projects requiring substantial resources need additional approvals directly from the executive committee.

The Architecture Review Forum is an IT-focused governance body engaging senior IT managers, enterprise, and solution architects. Participants of the ARF scrutinize the architectures of all proposed IT projects and assess their viability from the technology perspective. For instance, they review the main technical decisions taken by projects, validate them against the established standards and ideal future states described in technical reference architectures (when they exist), discuss potential deviations, and ensure that their architectures are as strategic as possible. Additionally, the Community Architecture Forum presents an opportunity for information sharing, idea dissemination, and communication to all architects. It has optional attendance and no formal governance authority.

FMCG has a flexible budgeting cycle that allows initiating and funding projects continuously over the year. Each project starts its life as an idea proposed by business stakeholders. These ideas can be either spontaneous or derived from the broader strategic direction defined by FMCG's executives. After an initial informal discussion and approval of the "seed" funding this idea is elaborated into a solution overview of the potential future IT project by an assigned solution architect. The solution architect engages relevant domain and subject matter experts and develops the solution overview based on the established standards and principles. Inventories providing the descriptions of currently available entities help solution architects reuse and leverage existing IT assets. For most areas technical reference architectures provide high-level descriptions of the current IT landscapes in these areas to facilitate the solution planning. Additionally, if the relevant technical reference architecture provides a description of the desired future state for the business capability that the project aims to enhance, then the solution architect aligns the solution overview to this ideal target state.

When the solution overview is ready, the solution architect prepares Key Design Decisions (KDDs) for the project and presents the solution overview together with its KDDs at the Architecture Review Forum (ARF) for discussion and consideration. he ARF reviews the solution overview and KDDs to ensure that the project is aligned to established principles, standards, and the target state defined in the technical reference architecture (if it is defined for the corresponding business capability), as well as to ensure that all potential deviations are justified. As a result of this review, the ARF concludes whether the project is desirable or feasible from the technical perspective.

After the solution overview is reviewed by the ARF, the business case for the project is prepared. A high-level description of the project provided by the solution overview is used as a basis for estimating its value, benefits, time, cost, and ROI that shape the business case. Then the business case, KDDs, and other documentation for the project are presented at the Innovation Forum where senior business and IT leaders make the ultimate investment decision on the project. Participants of the innovation forum consider three main factors when approving projects:

1. Financial considerations described in the business case
2. Alignment to the agreed IT investment roadmaps
3. Conclusions of the ARF on the technical desirability of the project

In certain cases the Innovation Forum can approve a project even if it deviates from the roadmaps or if it is not endorsed by the ARF; for example, when the project has compelling financial benefits, strict time limitations, or satisfies urgent legislative requirements. If the project is approved, then the business sponsor who initiated the project takes accountability for the benefits and outcomes estimated in the business case.

When the project is approved and funded, the solution architect with relevant domain and subject matter experts, develops a more detailed Solution Architecture Document (SAD) for the project and refines its KDDs. The SAD and KDDs are again reviewed by the ARF and then the SAD is passed either to an internal project implementation team or to a vendor in order to actually deliver the project. After the project is implemented, the solution architect conducts a post-implementation review in order to validate the delivery and verify the compliance to the KDDs and SAD. Project-level architecture processes at FMCG are shown in Figure 3.

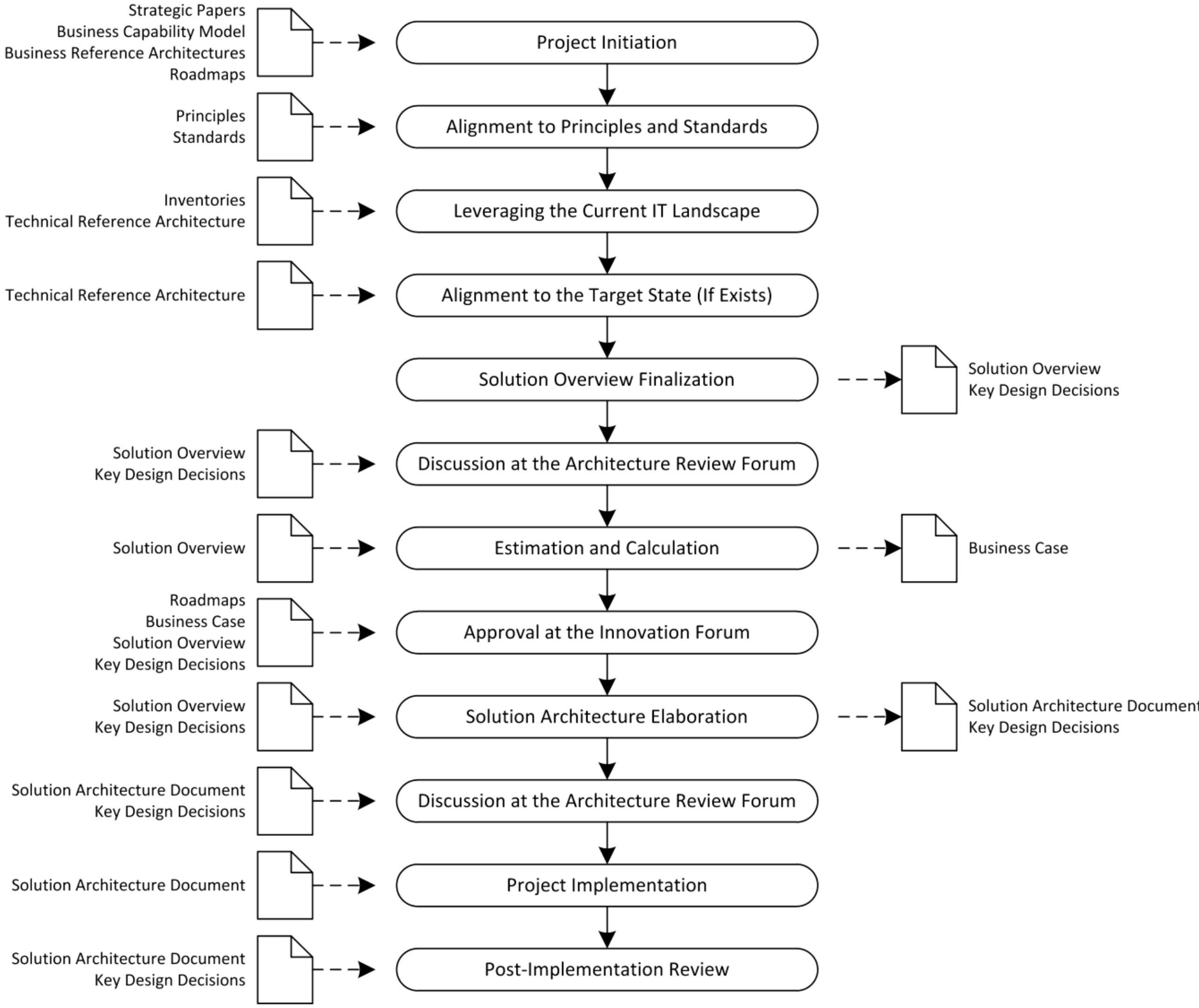


Figure 3: Project-Level Architecture Processes

PROBLEMS AND SOLUTIONS

As was discussed above, FMCG operates in the fast-moving consumer goods industry, which is very fast-paced, cost-sensitive, reactive, competitive, and rapidly changing. These specific features of the industry significantly influence the organization of the EA practice at FMCG. In this section I will discuss a number of industry-specific problems experienced by FMCG as well as their potential solutions, and mitigation strategies followed by FMCG. However, these solutions should not be considered as best possible, perfect, or final since the company is still looking for better ways of organizing its EA practice and alleviating these problems.

Unstable Business Strategy

Due to the rapidly changing external environment, the business strategy of FMCG is unstable and often shifts. This circumstance makes long-term architecture planning at FMCG problematic and poses a significant challenge for the EA practice. As described by a solution architect:

"In the traditional EA cycle with a plan for say three to five years, they [enterprise architects] posit a target state and perhaps an interim state, they'll create a roadmap for three to five years and they may spend 12 months getting that view, creating that view of the target and the roadmap. The problem with an organization like this is that in 12 months the organization has changed direction three or four times. So, you're not going to get that kind of stability that fits those timeframes. [...] An insurance company or a bank may have

the stability to be able to look five years ahead. In this industry things change, [...] it's constantly changing, it's very different."

FMCG follows a number of coping strategies to mitigate the negative effects of an unstable business strategy on the EA practice.

Long-Term Planning Is Focused on Essential Capabilities

Since the business strategy of FMCG is subject to constant change, the long-term architecture planning is focused on the key business capabilities essential for the organization regardless of any particular business strategy. As explained by an enterprise architect:

"We are operating at the capabilities that will always stay the same. For example, the ability to manage a product, we'll always need to do that. [...] But sometimes what happens is the priorities change. Well, today we were talking about the product information management, but now because something's happened with one of our products, they might change the focus and now we are introducing supply chains [...]. But the capabilities don't change, it's just a focus changes."

This approach largely resonates with the recommendation of Ross (2005) to "forget strategy" and focus architecture planning on the operating model, which reflects permanent business capabilies required by the organization and provides a more stable basis for planning than a busines strategy (Ross et al. 2006; Weill & Ross 2009).

Shortened Planning Horizon

Due to the unstable organizational environment, the architecture planning horizon at FMCG is reduced from the typical horizon of five years to three years. According to an enterprise architect:

"Traditionally it'll be a five years roadmap, but because our industry works up so quickly, things change so quickly, it's a three years [roadmap] and it's working to try to understand how technology will disrupt our market and what are the impacts of technology on some of the key factors that impact FMCG as a business."

Differentiated Planning

Since the long-term architecture planning at FMCG is troublesome, far from all business areas have their future states planned and the quality of these plans varies. For instance, more or less detailed roadmaps and business reference architectures exist only for a small number of the most essential capabilities, while tentative target technical reference architectures are planned for a wider scope of 20-30% most important business capabilities. Therefore, IT projects implemented in a certain business area tend to align to the best available description of the desired future state for this area, if any. As a solution architect described:

"In the absence of a formal roadmap each project will look at the business capability, will look at the reference architecture, will look at the target state to determine if we know in this particular domain where we're trying to go. Do we know the destination? Do we know the direction? If we do, to what extent is that direction and destination correct given what we now know about this particular project. And if it's fit for purpose, then the project will execute towards it. [...] We have a very patchy reference architecture, a very patchy view of what the target state is. In the absence of a known target state principles will organically guide the direction."

REACTIVE NATURE OF THE BUSINESS

FMCG's business is very reactive and fast-paced in nature. FMCG has to quickly respond to the monthly sales statistics, unanticipated competitors' moves, and recently legislated changes. These compelling factors often urge FMCG to implement "quick and dirty" IT solutions, which is problematic for the EA practice and even contradicts its general idea. As explained by a solution architect:

"As a general observation the fast-moving consumer goods industry [...] is a very reactive industry, things change very quickly. They report on a monthly basis, and depending on how things are going on a monthly basis funding [for projects] gets contracted or expanded. Therefore, it's very hard to fit that in with a traditional EA cycle."

According to the Manager of Architecture:

"Because FMCG and I'm sure all retail organizations are very fast-paced, they move very quickly, there isn't enough time to actually do a proper EA, there is no time. Business has moved even before you can say "go". They need something done very quickly. So, it's an interesting challenge."

As described by an enterprise architect:

"Sometimes what we get in this field is, for example, [a competitor] will do something or the government will introduce a new legislation, and then we need to do something to respond to that. Usually we just throw a solution architect [at it] and they'll sort that out."

FMCG employs several approaches to adapt its EA practice to the reactive nature of its business.

Future States are Implemented Opportunistically

Since FMCG is often forced to implement IT solutions addressing some urgent short-term needs, these solutions are typically considered as opportunities to get a step closer to the desired long-term target state. As explained by an enterprise architect:

"We're developing future states as we go. So, what we do is using the projects to fund future state creation. [...] Then a project will come along to deliver some new capability or enhance an existing capability and part of the solution architect's role is to [understand] what the future state will be.

They'll have a conversation with [enterprise architects], and [...] we will give them a view of what we are looking at doing X, Y and Z."

As described by a solution architect:

"[Technical] reference architecture is primarily used by the solution architects to basically guide their decisions. Ideally, when the target state is known, as we execute solutions or execute projects and develop solutions we are opportunistically trying to get towards the target state."

Bottom-Up Approach to Architecture

A multitude of relatively small but unexpected and important initiatives prevent FMCG from doing detailed top-down architecture planning. Instead, significant portions of FMCG's IT landscape are planned "just enough, just in time" with the significant involvement of solution architects, architecture emerges in a bottom-up manner.

As described by a solution architect:

"We're not good at doing top-down architecture. [...] The heavy lifting of architectural thinking is done by the solution architecture team. Therefore, things are done on a piecemeal basis, be it here or there. And we're trying create the pieces of the puzzle individually and then hope that they all fit together, or try to make them fit together. … We have to do EA differently in this organization, we're doing it organically, so to speak. That rises out of solution architecture."

Dispensations are Common

Due to a large number of IT projects of high tactical importance, architectural governance at FMCG is relatively loose and many solutions are allowed to deviate from agreed strategic roadmaps and planned long-term technical reference architectures. Important tactical projects are often given a "dispensation" by enterprise architects and proceed to implementation even when they are not fully aligned with the strategic vision.

As explained by an enterprise architect:

"[If a project is not aligned with the roadmap] we give the project a dispensation, [...] we're trying to be very pragmatic, we don't wanna be bureaucratic. The dispensation we give to the project would be based on ... that will have to demonstrate a reason why they have to go tactical. And mostly the reasons are that there is a business imperative; i.e., we need to do this because the governance said "we have to put labels to show a country of origin" and doing it in a strategic way [is a poor option] if we wanna meet the timelines. It's usually about timing. So, we just let it go ahead. But in other cases we would stop projects. … I wouldn't say [the target technical reference architecture] is strictly adhered to, it's used as guidance and there might be reasons, and they are usually financial or due to compliance, where we might issue a dispensation meaning that that project can go off the plans or become not referred to the reference architecture because of the need for specific domain or business requirements."

FOCUS ON A SHORT-TERM PAYOFF

FMCG's dynamic industry forces the company to focus more on getting tangible short-term returns on investment (ROI), rather than on uncertain longer-term benefits. In this "here and now" oriented culture the FMCG's EA team struggles to demonstrate short-term benefits and deliver the true strategic value for the organization.

As described by the Manager of Architecture:

"The organization is very much looking at what do we need to do today. So, for architecture there's a huge challenge to actually do something more strategic, more long-term, getting funding. You got to be clever in the way you actually look at strategic architecture."

FMCG uses a number of techniques to secure short-term benefits from all IT projects.

Large Projects are Split into Multiple Smaller Projects

Since FMCG is primarily looking for short-term payoffs, large IT projects are typically split into a number of smaller projects delivering some quick and tangible benefits. Each small project has its own solution architecture documentation and is approved individually. Ideally all IT projects should have an immediate payoff and still help to achieve certain strategic benefits at the same time. According to the Manager of Architecture:

"Usually [large projects] get delivered within the phases [...] and then we integrate them. [...] It's approved phase by phase."

Careful Estimation of Expected Short-Term Project Benefits

Due to its focus on getting short-term payoffs and benefits from IT projects, FMCG has a well developed capability of estimating these benefits and payoffs. Business cases for all projects should contain specific, quantitative, and measurable benefits that could be achieved in a reasonable timeframe. As a solution architect explained:

"What FMCG insists on, much more than other businesses, is when you make a business case you specify the benefits that you are likely to get out of this particular idea. Here there is an insistence on quantitative benefits. So, you need to actually show that it will increase sales by 1% in a particular channel or on a particular line of product or whatever. [...] Whatever you're claiming, you have to make it quantitative [...]. [And these benefits should be achieved] very soon, not after 5 years."

According to the Manager of Architecture:

"Each project should actually pay off. There's very few projects we've got where there's no commercial benefits."

Personal Accountability for Claimed Project Outcomes

Because of the importance of tangible short-term financial returns on investment (ROI), project sponsors at FMCG typically held personal accountability for the claimed benefits and outcomes of their projects. As described by a solution architect:

"Whatever you're claiming, you have to make it quantitative and someone in the business has to own the outcome. And if the project is approved on a basis of the benefits and costs that you've identified, then that business owner who owns that outcome will actually be held accountable for that outcome. If that benefit is not realized, they will be held accountable. So, it actually sharpens everyone's minds a little bit and focuses everyone. Because in a lot of other organizations benefits are very much hand-waving, "oh, yes, it will improve productivity, will do this, will do that". [...] But no one afterwards is actually saying "did we achieve that?" Here it's in more focus."

DRIVE TO REDUCE COSTS

Intense competition in the Australian fast-moving consumer goods industry creates a constant pressure for companies to reduce the costs required to run the business and makes cost-effectiveness one of the key objectives for FMCG. A continuous drive to lower the costs forces the EA practice at FMCG to maximize its benefits/costs ratio. As described by the Manager of Architecture:

"There's a lot of competition at the moment. [Competitor A] is coming, their prices are lower than ours. [Competitor B] is not doing very well at the moment, but they're trying to get there and don't give up. [Competitor C] is also coming to the market. So, it's all about keeping our costs down in terms of surviving as an organization. You can only keep your costs down if you lower your costs to run the organization. …It's commercializing architecture, I think that's what it is. If we've got a commercial lens on architecture I think it'll be very successful in an organization which is very conscious of cost."

FMCG adheres to two rules helping improve the cost-effectiveness of the EA practice.

Lean Architecture Function

In order to minimize the architectural overhead, FMCG tries not to inflate its architecture function beyond the necessary minimal size which is sufficient to deliver reasonable benefits for the organization. For instance, the manager of architecture admits that two enterprise architects currently employed by FMCG are focused mostly on the key critical areas and cannot handle the whole potential scope of architecture work. However, there are no plans to hire any additional enterprise architects. Therefore, the architecture function at FMCG is lean and essentially deliberately understaffed to maximize the value of each enterprise architect and avoid the effect of diminishing "marginal utility" of enterprise architects.

Pragmatism and 80/20 Rule

Driven by the desire to gain more with less, architects at FMCG tend to be pragmatic and follow an 80/20 approach towards architecture documents; i.e., develop and maintain only 20% of potentially desirable documents providing 80% of all potential benefits of using architecture. For instance, current technical reference architectures are not maintained for all areas, target technical reference architectures are planned only for important areas, while roadmaps exists only for the most critical areas. According to a solution architect:

"We do have a business capability model. Is it 100% accurate or current? Probably not, but it's fit enough for purpose. [...] I'm a great believer in the 80/20 rule, near enough is good enough for most purposes. And in business that's a general principle here. … You will find that organizations like FMCG in this particular industry are very pragmatic. It's all about what makes a difference in terms of cost and revenue. It's less about aesthetic purity or architectural purity or consistency than about just getting the job done."

SHORTAGE OF ARCHITECTS WITH RELEVANT EXPERIENCE

Due to a number of industry-specific features of the EA practice at FMCG described above, the company experiences problems with finding architects acquainted with the specifics of the fast-moving consumer goods industry on the job market. This problem does not have any specific solutions at FMCG. As explained by a solution architect:

"Because people are coming from other more traditional organizations, they're trying to plan ahead too far and take too long to do it."

CONCLUSIONS

In this article I discussed the EA practice in a large Australian retail chain operating in the fast-moving consumer goods business and the impact of industry on this EA practice. The analysis provided above clearly suggests that specific features of a particular industry can significantly influence the approaches to an EA practice followed by the organizations operating in that industry. As described by a solution architect:

"I think really what is required here is that people have to move away from heavy EA, there's just no space for it here. [...] The timeframes here are so short and a funding cycle is quite fluid and fluctuates a lot. You don't have the luxury of being able to spend 6 or 12 or 18 months to come up with a five year roadmap, that's never going to work here, ever. [...] EA needs to show much more direct value. For example, if you think about traditional EA with five year periods, that return on investment will never be realized here."

According to the Manager of Architecture:

"Other organizations, like banks, they're little more regulated, they're more slow-paced. So, I think EA at FMCG and at retail is gonna take a very different flavor [...]. [The] TOGAF® [framework] doesn't quite fit in here."

As explained by a solution architect:

"It's not that [the architects] don't understand how to do [EA] properly, it's that the definition of properly in terms of the broader industry is not fit for purpose here."

The fast-moving consumer goods industry is very fast-paced, competitive, cost-sensitive, and reactive. These industry-specific factors profoundly shaped the EA practice at FMCG, which has undergone a long evolutionary process striving to adapt to the nature of industry. The resultant EA practice can be described as "lean", "loose", and "agile", favoring efficiency over effectiveness. However, the evolution of the EA practice at FMCG is far from finished and the company is still looking for better ways to position and leverage EA. As explained by a solution architect:

"We had discussions around what does EA mean for FMCG, how could an EA practice work here. And I guess an EA practice here would have to be far more responsive, more agile, and more tuned to the nature of industry. In other words, the approach taken for EA can't be one-size-fits-all across all industries."

According to the Manager of Architecture:

"We've got to redefine what EA means at FMCG. It's got to be more fit for purpose."

And as described by a solution architect:

"We're reassessing what EA means and how we do EA. How do we have a roadmap for an organization that's constantly changing direction? What does having a roadmap mean?"

Unfortunately, the current EA theory and popular approaches to EA do not provide any meaningful answers to the questions discussed in this article. Therefore, the gaps in our knowledge regarding the specifics of an EA practice in different industries open fruitful directions for future EA research of high potential importance to both EA theory and practice.

ABOUT THE AUTHORS

Svyatoslav Kotusev (kotusev@kotusev.com) is a researcher at RMIT University, Melbourne, Australia. He has spent the last three years studying EA practices in organizations. Prior to his academic career he held various software development and architecture positions in industry. He is a TOGAF® Certified expert.

Mohini Singh is a professor in the discipline of Information Systems at RMIT University. She has to date published more than 150 peer-reviewed papers in the areas of IT and Innovation Management, e-Business and e-Government, e-Services, and Social Media. She has successfully completed several funded research projects supported from category A funding schemes.

Ian Storey is a lecturer in the Business Information Systems and Logistics department at RMIT Melbourne. His PhD was on the topic of genetic algorithms and multi-objective optimization. Research interests include information security, EA, Control Stabilization, and eLearning. He received a BSc from Queensland University and a Masters in Information Security from RMIT.

REFERENCES

F. Ahlemann, E. Stettiner, M. Messerschmidt, C. Legner (Eds.): Strategic Enterprise Architecture Management: Challenges, Best Practices, and Future Developments, Berlin: Springer, 2012.

P. Anderson, G. Backhouse, J. Townsend, M. Hedges, P. Hobson: Doing Enterprise Architecture: Enabling the Agile Institution, 533, Joint Information Systems Committee (JISC), Bristol, United Kingdom, pp.1-82, 2009.

S. Gerber, U. Meyer, C. Richert: EA Model as Central Part of the Transformation into a More Flexible and Powerful Organization, in Proceedings of the 2nd Enterprise Modeling and Information Systems Architectures Workshop, M. Reichert, S. Strecker, K. Turowski (Eds.), St. Goar, Germany: Gesellschaft für Informatik, pp.23-32, 2007.

S. Gonzalez: An Enterprise Architecture for Banking, Journal of Enterprise Architecture (7:4), pp.70-79, 2011.

P. Hungerford: The Syngenta Architecture Story, in Handbook of Enterprise Systems Architecture in Practice, P. Saha (Ed.), Hershey, PA: Information Science Reference, pp.331-350, 2007.

P. Hungerford: The Evolving Role of Enterprise Architecture within Syngenta, in Coherency Management: Architecting the Enterprise for Alignment, Agility, and Assurance, G. Doucet, J. Gotze, P. Saha, S. Bernard (Eds.), Bloomington, IN: AuthorHouse, pp.307-328, 2009.

S. Kotusev, M. Singh, I. Storey: Consolidating Enterprise Architecture Management Research, in Proceedings of the 48th Hawaii International Conference on System Sciences, R.H. Sprague (Ed.), Kauai, HI: IEEE, pp.4069-4078, 2015.

N. Lynch: Enterprise Architecture – How Does it Work in the Australian Bureau of Statistics?, in Proceedings of the 17th Australasian Conference on Information Systems, S. Spencer, A. Jenkins (Eds.), Adelaide, Australia: Association for Information Systems, pp.1-14, 2006.

S. Murer, B. Bonati, F.J. Furrer: Managed Evolution: A Strategy for Very Large Information Systems, Berlin: Springer, 2011.

Y. Park, G. Lee, Z. Lee: EA Configurations: Interplay of EA Design Factors, Strategy Types, and Environments, in Proceedings of the 34th International Conference on

Information Systems, R. Baskerville, M. Chau (Eds.), Milan, Italy: Association for Information Systems, pp.1-9, 2013.

T.E. Pheng, G.W. Boon: Enterprise Architecture in the Singapore Government, in Handbook of Enterprise Systems Architecture in Practice, P. Saha (Ed.), Hershey, PA: Information Science Reference, pp.129-143, 2007.

A. Rees: Establishing Enterprise Architecture at WA Police, Journal of Enterprise Architecture (7:3), pp.49-56, 2011.

G.L. Richardson, B.M. Jackson, G.W. Dickson: A Principles-Based Enterprise Architecture: Lessons from Texaco and Star Enterprise, MIS Quarterly (14:4), pp.385-403, 1990.

J.W. Ross: Forget Strategy: Focus IT on your Operating Model, Center for Information Systems Research, MIT Sloan School of Management, Cambridge, MA, 2005.

J.W. Ross, P. Weill, D.C. Robertson: Enterprise Architecture as Strategy: Creating a Foundation for Business Execution, Boston, MA: Harvard Business School Press, 2006.

P. Saha: The Four Design Models of Enterprise Architecture, in Coherency Management: Architecting the Enterprise for Alignment, Agility, and Assurance, G. Doucet, J. Gotze, P. Saha, S. Bernard (Eds.), Bloomington, IN: AuthorHouse, pp.51-76, 2009.

H.A. Smith, R.T. Watson, P. Sullivan: Delivering an Effective Enterprise Architecture at Chubb Insurance, MIS Quarterly Executive (11:2), pp.75-85, 2012.

V. Venkatesh, H. Bala, S. Venkatraman, J. Bates: Enterprise Architecture Maturity: The Story of the Veterans Health Administration, MIS Quarterly Executive (6:2), pp.79-90, 2007.

P. Weill, J.W. Ross: IT Savvy: What Top Executives Must Know to go from Pain to Gain, Boston, MA: Harvard Business School Press, 2009.

Short Subject

Guiding Principles to Support Organization-Level Enterprise Architectures

Aaron Trionfi

Abstract

Enterprise Architecture (EA) practices have long served as the foundation for information technology solution development. Most EA methods and frameworks claim that these same practices can be applied to the development of an EA for an entire organization, but attempts to develop architecture on this scope routinely fail. The author contends that EA practices and frameworks must be extended to better implement organization-level architectures. An EA program should follow four principles when attempting to develop an organization-level architecture: a strong metamodel over a strong product catalog, business intelligence over modeling, integrated data capture over data calls, and data quality over data quantity.

Keywords

Metamodel, business intelligence, data, organization-level architecture, portfolio management

INTRODUCTION

Enterprise Architecture (EA) means different things to different people. EA frameworks such as the Zachman Framework, The Open Group TOGAF® standard, Federal Enterprise Architecture Framework (FEAF), and Department of Defense Architecture Framework (DoDAF) acknowledge that EA has multiple levels of stakeholders, each of which has different expectations of and needs for EA. Said another way, each level of stakeholder expects different questions to be answered by the EA. Three common "levels" of EA – each having a different stakeholder group – are the enterprise or organization-level architecture, the segment or functional area architecture, and the solution architecture.

Many of the commonly adopted EA frameworks, especially in the US federal government, involve practices that tend to work very well at the solution level, but fall short as the scope of the architecture increases to the organization level. One of the biggest weaknesses is the idea that architecture work must be captured or conveyed as pictures, diagrams, and models. The use of such architecture products (to use DoDAF terminology) for organization-level architectures has four main problems:

- Standard EA products become too complicated as the scope expands, requiring too many resources to produce or maintain as well as taking too long to create. By the time the products are complete, they are already outdated and have cost the organization too much.
- Creating and reading most EA products require special skill sets, not commonly held throughout the enterprise. Consequently, the information captured in EA products cannot be conveyed quickly, especially to executive-level decision-makers.
- The information displayed in the EA products is done so in an unstructured way, making computer-aided analysis of it almost impossible.
- Few, if any, modeling standards are well accepted for higher-level architecture elements like strategies and goals.

In a 2008 study, Jonathan Broer found that roughly 66% of surveyed organizations indicated that their EA programs did not achieve the expected results (Roeleven et al. 2009). The weaknesses of current EA frameworks mentioned earlier likely contribute to many failing EA programs as does the belief of many EA programs that all the questions they need to answer can be answered using common EA frameworks. The use of frameworks that do not support communication with non-architects and cannot support the needed analysis reduces the potential value of EA programs to other functional areas of the organization.

To address the observed weaknesses of current EA frameworks, this article proposes a set of principles to extend common EA frameworks to better support organization-level architecture development. They are:

- A strong metamodel over a strong product catalog
- Business intelligence over modeling
- Integrated data capture over data calls
- Data quality over data quantity

The principles use a structure very much like the tenets described in the Agile Manifesto. Although the elements on the left in each principle are more valued, the elements on the right are also recognized to have value. In this case, the elements on the left present greater value when attempting to develop organization-level architectures. The sections that follow elaborate on the four principles.

PRINCIPLES FOR ORGANIZATION-LEVEL EA

1. A Strong Metamodel over a Strong Product Catalog

Most EA frameworks include a metamodel; i.e., a model of the concepts used by the framework. These metamodels, however, tend to be overshadowed by the products or views prescribed in the framework. Some aspects of the metamodel will not be relevant to solution-level or even segment-level architectures. Still others are poorly represented or neglected in the most commonly used EA products. This can result in failure to populate some areas of the metamodel. Conversely, in some frameworks, the product catalog is used to determine the elements of the metamodel. In essence, the EA metamodel becomes a data model representation of product elements. While this approach may be sufficient for compiling solution-level architectures, it breaks down at the organization level due to the issues mentioned earlier (complexity of products, resources required, etc.).

For organization-level architectures, I suggest focusing on the EA metamodel, which should be independent of any potential EA products being developed at the solution level. Instead, the metamodel should be developed with a core set of analytics in mind (such as common metrics used for investment portfolio management). By not strictly aligning solution-level EA products to the EA metamodel, development teams will have the freedom within the confines of the adopted EA framework to optimize the documentation of their solution design.

With a focus on the EA metamodel, the EA repository must extend beyond a document store for EA products. It must act more as relational database. However, the document store should still be available, and ideally use many of the object classes in the database as ways to categorize the stored documents.

The EA metamodel is likely to be developed and visualized using a Unified Modeling Language (UML®) class diagram or entity-relationship diagram. Therefore, communicating with executives and other EA stakeholders and educating them on the EA metamodel will be difficult. I suggest developing a conceptual model version of the metamodel with many of the more complicated elements stripped out. Once the model is created, the communications should revolve around how linking the different data areas can allow the EA program to answer business questions pertinent to stakeholders. Figure 1 shows an example of a simplified metamodel, and Table 1 defines the data classes.

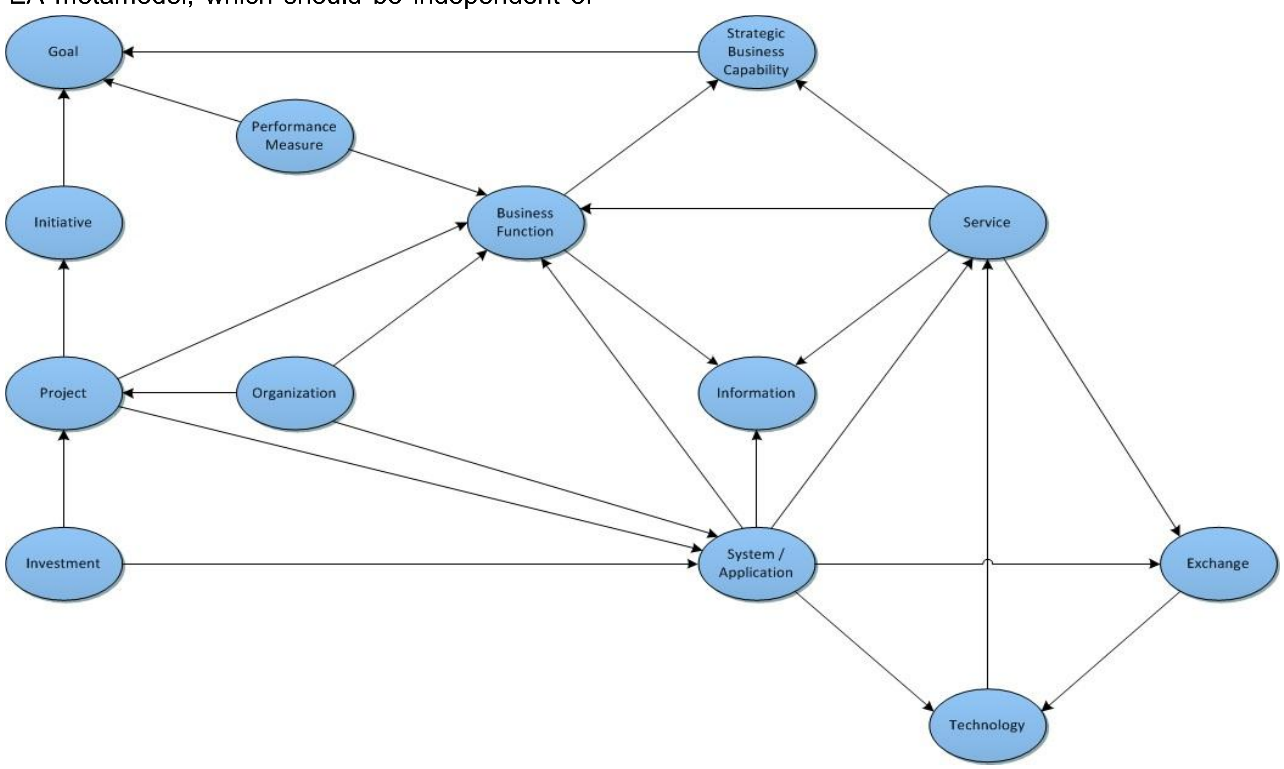


Figure 1: A Simplified EA Metamodel Describing the Objects of the Organization about which an EA Program may Collect Data

The image can be used to more easily communicate the data needs of the EA program and how the data can be integrated to answer business questions. This example was used in an EA program focused on organization-level architecture and its use to better manage strategic information technology (IT) investments.

Table 1: Definitions of the Elements in the Example Metamodel Shown in Figure 1

Data Class	Definition
Goal	An end result whose success can be measured.
Initiative	A program or group of projects that help achieve a goal.
Project	A timebound set of activities that result in a tangible output that helps accomplish a goal.
Investment	Funding used to help imrove or maintain an organization's operations.
Performance Measure	A means of quantifying the success or failure of goals as well as day-to-day operations.
Organization	The organizational structure of the enterprise, including organizational divisions, people, positions, and roles, as appropriate.
Business Function	The operations performed regularly by the organization to achieve its strategic business capabilities.
Strategic Business Capabilities	The primary purpose of the organization (synonomous with mission).
Information	Both the structured data and the unstructured information used by the organization.
System/Application	The deployed technology used by the organization to execute its business functions.
Service	An operation performed by a provider for the benefit of the receivers (typically in exchange for a fee).
Technology	The foundational software, hardware, and protocols used by the organization to build its systems/applications and provide basic IT services to its personnel.
Exchange	A means of transferring information or data.

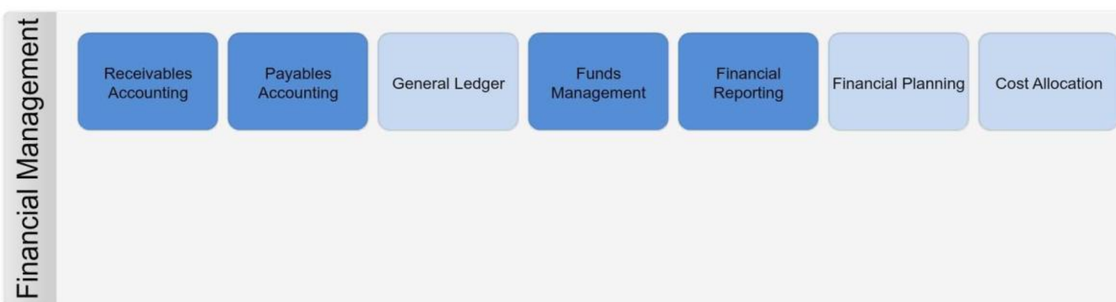


Figure 2: A Heat Map Showing the Business Functions of the Financial Management Capability

The heat map is color-coded by the number of business applications supporting those functions. Darker blue represents more business applications.

The communications using the metamodel must be tailored to each audience. For instance, for financial executives, the communications might focus on how linking investment data to systems/applications can help determine how a reduction in specific investments will impact the maintenance of current business systems. Conversely, when for a functional office providing services, the communications could focus on understanding the impact on their services if the staff executing a specific business function is reduced.

By focusing the organizational architecture on populating a metamodel, the architects developing it can work strictly on the collection of important data rather than on developing, in many cases, unwieldy models that cannot be completed in a reasonable period or be consumed by those making organization-level decisions. The metamodel also becomes much easier to adapt as analytic requirements change because alterations do not require corresponding modifications to EA product specifications.

Finally, by focusing on the collection of data to populate the EA metamodel, the architect will compile a set of reference data (a set of business functions, a system inventory, organization goals, and technology components, for example) that can be used as building blocks, as defined in the TOGAF standard, for solution and segment-level architectures as well as a means to categorize content, including solution and segment architectures.

2. Business Intelligence over Modeling

EA programs commonly help decision-making by identifying operational and IT redundancy, finding gaps, and other such analyses. At the organization level, this means the targeted audience for such analyses is likely senior executives. Reports to such an audience usually begin with an executive summary, which is typically all an executive will read due to time constraints. The details are available in the body, however, if the executive wants them. More times than not, organizational EA programs fail to connect with their audience (senior executives) because they cannot effectively summarize EA information that historically is conveyed through EA models.

The growth of the business intelligence field gives EA programs a powerful set of tools to help summarize typical EA data in ways that senior executives can more quickly consume, while helping the EA team perform some of the standard analyses for which it is responsible. Figure 2 shows an example.

Figure 2 summarizes the number of applications that support specific business functions related to financial management. This visual can help a senior executive determine whether specific functions need to be examined further for potential IT redundancies. In comparison, this information would be conveyed using a matrix (SV-5) in DoDAF. Such a matrix would show a list of applications and functions and force the audience to perform the summation needed to identify the functions with higher redundancy risk.

Other common business intelligence reports include standard chart types (pie, scatter, bar, or bubble), road maps (Gantt-style charts), and meter/dial charts.

These business intelligence reports represent a new type of EA product or view. When choosing or defining the organization's EA framework, the EA program should document these business intelligence reports in a manner similar to the more traditional, model-based products. However, the EA framework should allow the organization-level EA practitioners enough freedom to develop new business intelligence reports when needed.

Using business intelligence summarization techniques, the EA program can significantly improve its ability to communicate effectively and rapidly with its most important stakeholders. It can also position itself as a key enabler of data-driven decision-making in the organization – a highly coveted capability, especially in federal government agencies.

3. Integrated Data Capture over Data Calls

A common approach to constructing an organization-level architecture is to perform large, manual data calls to compile the data necessary for the products being created. For instance, a team of enterprise architects may build a data template (something in Microsoft Excel®, for example) and send it to a large number of subject matter experts (SMEs) to complete. The creation of the template can be time-consuming, as will the effort to ensure it is properly filled out by the SMEs. However, in most cases, this data is already being managed elsewhere in the organization. Table 2 shows some examples.

Table 2: Data Categories (typically managed by functional areas other than EA) Used to Create an Organization-Level Architecture

Data Category	Functional Area
Organization	Human resources
Goals	Executive team (strategic plans)
System/Application	Information security or development shops
Technology	Network management, server management, information security (white list)
Project	Project management office
Investment	Financial management

The organization may already have well established processes and applications in place for managing changes to the information for each of the examples in Table 1. An organizational EA program should have the goal to integrate itself into these processes and applications. The EA program can offer to add value to the functional areas in exchange for a "seat at the table" on relevant decision-making bodies and access to data sets. For instance, the EA team could offer to align projects with functional areas and help the project management office determine whether redundant or unnecessary projects are being executed. This type of analysis is only possible with integrated data structured on the basis of the EA metamodel. Another way for the EA program to add value is when the functional area does not have a strong application to manage its data. In this case, the EA program could offer the EA repository as a means to better manage and store the functional area's data.

This proposal reflects sound data management practices regardless of the type of data collected. A data collector should always attempt to identify an authoritative source of data. Such sources have a defined responsibility to ensure their data meets the six core data quality dimensions: completeness, uniqueness, timeliness, validity, accuracy, and consistency. By federating the data quality responsibilities, the EA team can better focus on integrating different data sets and performing analysis instead of constantly performing data quality assurance on individual data sets.

An EA program *should not* dismiss data calls altogether. Data is likely to be needed for an organization-level architecture that is not being formally managed elsewhere in the organization. In such cases, a data call may be the only means by which the data can be captured, but the EA program should make every effort to determine how to properly manage the data collected to avoid future data calls asking for the same data.

The key to this principle is to minimize the work functional teams need to do to provide needed EA data. This will lower the cost-benefit ratio the functional teams face for providing it. This, in turn, is likely to improve the sometimes contentious relationship between the EA program and its functional counterparts.

4. Data Quality over Data Quantity

At the organization level, the amount of EA data collected can easily grow out of hand, creating multiple problems. First, as the amount of data collected increases, the required effort to ensure data quality also increases. Second, because most data comes from other parts of an organization, a large data request can discourage the data provider from providing the data or ensuring its quality. This can come in the form of resisting providing interfaces to data sets or failing to respond to data calls.

As recommended in many architecture frameworks, data needs should be defined by the products that will make up the architecture. In the case of an organizational architecture, those products should be the business intelligence reports discussed in principle 2. Instead of viewing data collection efforts as an opportunity to collect "everything it can", the EA program should concentrate the collection work on only the data needed for the identified business intelligence reports.

The EA program should focus on ensuring the quality of the data collected rather than the amount. As discussed in principle 3, some of this responsibility should fall to the data providers. However, the EA program needs to ensure the data is being collected from the proper sources and that it is being collected at the right frequency.

One way to improve data quality is to provide enough detail in the EA metamodel to define what data quality means. This may include defining data types for fields, defining cardinality of relationships between metamodel classes, indicating update frequency for fields, and other such details. Ensuring data quality also means establishing auditing practices to routinely measure the quality of the data. This could include looking for data not compliant with the rules defined in the metamodel as well as looking for data elements that have not been updated for an extended period.

Without quality data, the business intelligence reports developed by the EA program risk being incorrect or incomplete. Presenting such reports to executive leadership can result in misinformed decisions and, eventually, a diminishing trust in the EA program. Conversely, business intelligence based on quality data can help decision-makers make better decisions, leading to a stronger trust in the EA program.

CONCLUSION

The principles above represent a significant modification to common EA frameworks such as the TOGAF standard, DoDAF, and FEAF. These modifications, if adopted by organization-level EA programs, should benefit the organization as follows:

- EA work will be easier to complete because creating large, complex models and diagrams for the organization-level architecture will not be needed.
- The need for personnel with advanced modeling and diagramming skills will be reduced, making EA programs easier to staff.
- The products produced for the organization-level EA will be consumable by more than just other architects, providing value to other functional areas of the organization.
- For US civilian government agencies, the data collected can facilitate responses to the Office of Management and Budget for compliance reporting such as those for PortfolioStat, the Federal Information Security Management Act, and capital planning and investment control.

- A set of reference data will be generated that can be used for categorizing segment and solution architectures as well as other content throughout the organization.

These ideas are directed toward EA programs developing organization-level architectures. As the scope of an architecture is reduced, the more traditional, model/diagram-based approaches to EA have more value and the principles presented here will become less important. However, by viewing these principles as a complement to current EA frameworks and approaches – as opposed to a replacement for them – EA programs can improve their ability to execute their mission and influence top-level decision-makers in their organizations.

ABOUT THE AUTHOR

Aaron Trionfi is a staff member of LMI's Enterprise Architecture team. He has supported US government agency EA programs for approximately six years. Over that time, he has developed architectures using multiple frameworks and every layer of architecture. Dr. Trionfi earned a PhD in physics from Rice University and uses this foundation to bring a strong analytic approach to EA.

REFERENCES

K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. Martin, S. Mellor, K. Schwaber, J. Sutherland, D.Thomas: The Agile Manifesto (2001); retrieved from http://agilemanifesto.org.

DAMA UK Working Group: The Six Primary Dimensions for Data Quality Assessment (2013); retrieved from www.damauk.org/RWFilePub.php?&cat=403&dx=2&ob=3&rpn=catviewleafpublic403&id=106193.

Officer, DoD Deputy Chief Information: Department of Defense Architecture Framework, Washington, DC (2010); retrieved from http://dodcio.defense.gov/dodaf20.aspx.

Federal Enterprise Architecture Framework (FEAF), Washington, DC; retrieved from www.whitehouse.gov/omb/e-gov/fea.

S. Roeleven, J. Broer: Why Two Thirds of Enterprise Architecture Projects Fail, ARIS Expert Paper (2009).

TOGAF®, an Open Group Standard (2011); retrieved from http://pubs.opengroup.org/architecture/togaf9-doc/arch/.

US Federal Government: Federal Information Security Management Act (44 U.S.C.§ 3541) (2002).

J.A. Zachmam: A Framework for Information Systems Architecture, IBM Systems Journal, 26, 454-470 (1999).

Short Subject

Next Gen Architecture – IT Trends and the API Effect

Michael Hinnebusch

Abstract

On the subject of the digital economy, much is written about API management, also known as "API-ification". Each Autumn, Gartner makes their annual predictions on IT trends impacting businesses for the upcoming year. Their theme in 2016 centered on an idea which it calls the "digital mesh". This article brings together the concepts of digital mesh and APIs to highlight their important effects on business today. Organizations need to comprehend how they relate to their customers' digital experiences. The article includes thought-provoking questions to challenge leaders to look toward the digital horizon and take action. Included is a discussion and a call to action. If followed, the result will provide a foundation for an organization's digital operating model.

Keywords

API-ification, digital operating model, application program interfaces, device mesh, user experience, information of everything, advanced system architecture, mesh app, service architecture, internet of things, IoT, ecosystem, value proposition canvas, customer digital story, value proposition, business model canvas, execution model

INTRODUCTION

This article draws together insights mainly from two different sources to aid business leaders' understanding of APIs and describe their increasing effect on the digital landscape of their industry. One source is from the well known research and advisory firm Gartner which provides information technology-related insight. The other comes from The Center for Global Enterprise (CGE), a private, non-profit, non-partisan research institution devoted to the study of the contemporary corporation, globalization, economic trends, and their impact on society (CGE 2016).

In the business setting, API is an acronym that stands for application program interfaces. APIs are created by software engineers to allow applications to communicate with each other. Architects and developers are finding new uses for APIs, and they are using them to replace cumbersome and antiquated means for sharing data between applications.

Much is being written about APIs from a technology and business perspective. Like most new technologies, API technology is going through its hype cycle. With the prospect of information overload, and the perceived technology hype cycle, leaders may be inclined to ignore the clamoring around APIs. This article highlights essential information on the subject and demonstrates the opportunity that business leaders must face.

THE DIGITAL MESH

In describing the Top 10 Strategic Technology Trends for 2016, Gartner introduces a concept which it calls the "digital mesh". This digital mesh ties together the Internet of Things, smart devices, and the evolving digital business (Gartner 2015). Interestingly, the "digital mesh" theme bridges six of the Top 10 Trends for 2016:

- The Device Mesh
- Ambient User Experience
- Information of Everything
- Advanced System Architecture
- Mesh App and Service Architecture
- Internet of Things Platforms

These strategic technology trends, woven together, may significantly alter markets and industries. But what ties together these products, strategies, and technology? It is the many Application Program Interfaces (APIs) that literally link together the growing digital mesh.

APIs can be either open or closed. Open APIs are available to the public and may be leveraged by a multitude of developers. Closed APIs are locked to the public and require permission to use. As an example, retailers may use open APIs for product information to enable more e-commerce. Online payment services will lock their APIs closed to protect sensitive financial transactions.

Before further discussing APIs, the following section describes Gartner's Top 10 Technology Trends for 2016. As in most years it is an interesting read and it provides more foundation for the API discussion.

Gartner's Top 10 Strategic Technology Trends for 2016 (Galer 2016)

Emerging technologies are poised to break out and disrupt business models in 2016. According to research firm Gartner, unprecedented connectivity between a plethora of devices tapping intelligence from smart machines will drive this dramatic transformation. The results include dazzling user experiences and business advances. David W. Cearley, Vice-President and Gartner Fellow, said the shift to digital business is the fundamental theme behind next year's top technology trends. During a recent Gartner webinar, he identified three overarching topics within the top 10 trends supporting this move: the physical and virtual worlds merge, intelligence everywhere, and all applications become cloud-centric. Cearley said:

"Digital business is evolving to become algorithmic business, which focuses on action. We're encapsulating data from activities into algorithms, creating more smart and intelligent systems as part of the entire digital business."

1. **Device mesh** is the expanding, pervasive technology underpinning that delivers frictionless, fluid, dynamic connections involving people, things, and businesses whether you're with a customer, working in a warehouse or an oil field, shopping in a retail store, eating at a restaurant, watching the game, or driving in your car. According to Cearley:
   "Smartwatches are the tip of the iceberg. Many of these technologies are very targeted so you need to think about certain opportunities like notifications, micro-interactions, and different levels of control or context that's provided to the user, such as employee productivity with authentication mechanisms or hands-free production."
   As one example of the growth of augmented and virtual reality, Gartner predicted head-mounted displays will dramatically increase from 260,000 unit shipments in 2015 to 2 million units next year and 25 million units by 2019.
2. **Ambient user experience** seamlessly blends the physical and virtual, adapting contextual information including user actions, environmental sensors on available devices, historical data, and capabilities from applications. Cearley said this will evolve to a unified multi-sensory experience:
   "Application design has to think outside-in starting with the user. You're designing for the personal cloud that exists for the user – their devices and scenarios that change throughout the day."
3. **3D printing** using advanced, multiple materials will create high-value innovation opportunities in industries like pharmaceuticals, life science (prosthetics and skin), electronics, food, and industrial manufacturing.
4. **Information of everything** will begin to make data meaningful to create smart machines that store, collect, and share valuable, actionable knowledge sources across the business ecosystem.
5. **Advanced machine learning** uses new types of models to infuse greater intelligence into systems. This is the next step in analytics, functioning as the "brain" of autonomous, smart machines able to learn, act, and adapt behavior.
6. **Autonomous agents** and things are on the rise creating new business opportunities. This includes robots, drones, driverless vehicles, virtual customer and personal assistants, smart appliances and tools, smart security and operations, and smart enterprise apps.
7. **Adaptive security architecture** goes beyond blocking the company's virtual perimeter to building risk mitigation into the system on a continuous basis using advanced behavioral and entity analytics that predict, prevent, investigate, contain, and remediate incidents.
8. **Advanced systems architecture** supports smart machines, in which specialized appliances can dramatically scale connections, turbo-charging innovations like facial recognition and cognitive learning.
9. **Mesh app and service architecture** emerges as a unified model to build microservices linked together into applications delivered across devices in the digital mesh. Cearley cautioned:
   "There's a significant learning curve and discipline required for this. It's not for everyone today but is one of the hottest topics."
10. **Internet of Things architecture and platform** has to be built so all the components are linked together including analytics, orchestration, data, an integration layer, aggregated device management, gateways, and the user interface.

[Taken from digitalistmag.com, December 2015]

THE DIGITAL MESH THROUGH THE API LENS

Application Program Interfaces (APIs) allow computers and IT services to exchange data. APIs are being created continuously, with added functionality, and empowering companies to provide new and more valuable business capabilities.

In September 2015, Peter C. Evans and Rahul C. Basole published a visual representation of data across thousands of APIs, and tens of thousands more of mashups, covering hundreds of categories ranging from search and e-commerce to transportation, health, and enterprise. They converted the API data into a visual network representation. Nodes represent APIs and links between the nodes represent if two APIs have been used jointly in a mashup. Links are scaled according to the total number of mashups: the thicker the line the more mashups were created using the two APIs. The result is an imagining of the current API economy presented on the following page. This is not "The digital mesh", but it begins to shed light on the concept and give a place for the business to attach its ideas (Basole & Evans 2015).

This API image is exciting, like looking up to the sky and seeing the stars on a clear night. Just as early astronomers could not have imagined space exploration and a network of manmade satellites, our future vision is limited as well. However, today we begin to draw parallels to these discoveries and take our steps to understand our companies' future digital operating model.

Basole and Evans delved into the data and presented another view to relate the online retailer Amazon with the big box retailer Walmart. See the resulting image on the following pages, "API Economy Visualized: Amazon *versus* Walmart".

The resulting image is thought provoking. It is not the aim of this article to make predictions on success of these two companies. However, it seems that the giant retailer, with all of its resources, is at a striking disadvantage in the digital economy.

What insights can you take from this pertaining to the retail industry? How can the big-box retailer, with its e-commerce API, compete against Amazon which is into so many other digital pathways such as social media, job search, tools/cloud/big data, enterprise/storage, payments, and messaging services?

Should other retail companies have a competing strategy, or a tactic to partner with Amazon? Can any other retailer legitimately compete in the digital space or should they be considering how to join the API economy led by Amazon?

What would an image for your industry API digital mesh look like? How would it appear for the top industries such as health technology, finance, technology services, or electronic technology? What should it look like a few years from now?

In the following section we will discuss what your company can be doing to get tied in with today's digital mesh.

TODAY'S DIGITAL MESH AND YOUR FUTURE

Organizations ordinarily do not create an industry ecosystem or its digital mesh. Typically they take part in its activities. Organizations play a role and have varying degrees of influence or impact on their own network. A company typically does not manage their ecosystem. However, to survive and thrive companies need to stay ahead of the trends, be familiar with the leaders in their industry ecosystem, know what they can do to gain favorable position within it, and clearly market and deliver on their core value proposition for customers.

The following outline a set of vital activities to begin:

- Ecosystem map (Who is involved?)
- Value proposition canvas (What is its worth?)
- Customer digital stories (Why do they engage?)
- Digital mesh (How does it fit together)
- Business model canvas (How to deliver?)
- Execution model (Sequence to take action?)

These steps will help define the current state and offer a glimpse of what is ahead for the digital mesh. The organization needs to fully understand the current state and what opportunities lay just ahead. Who are the players in your industry ecosystem? Who are the leaders, decision-makers, influencers, followers, and collaborators?

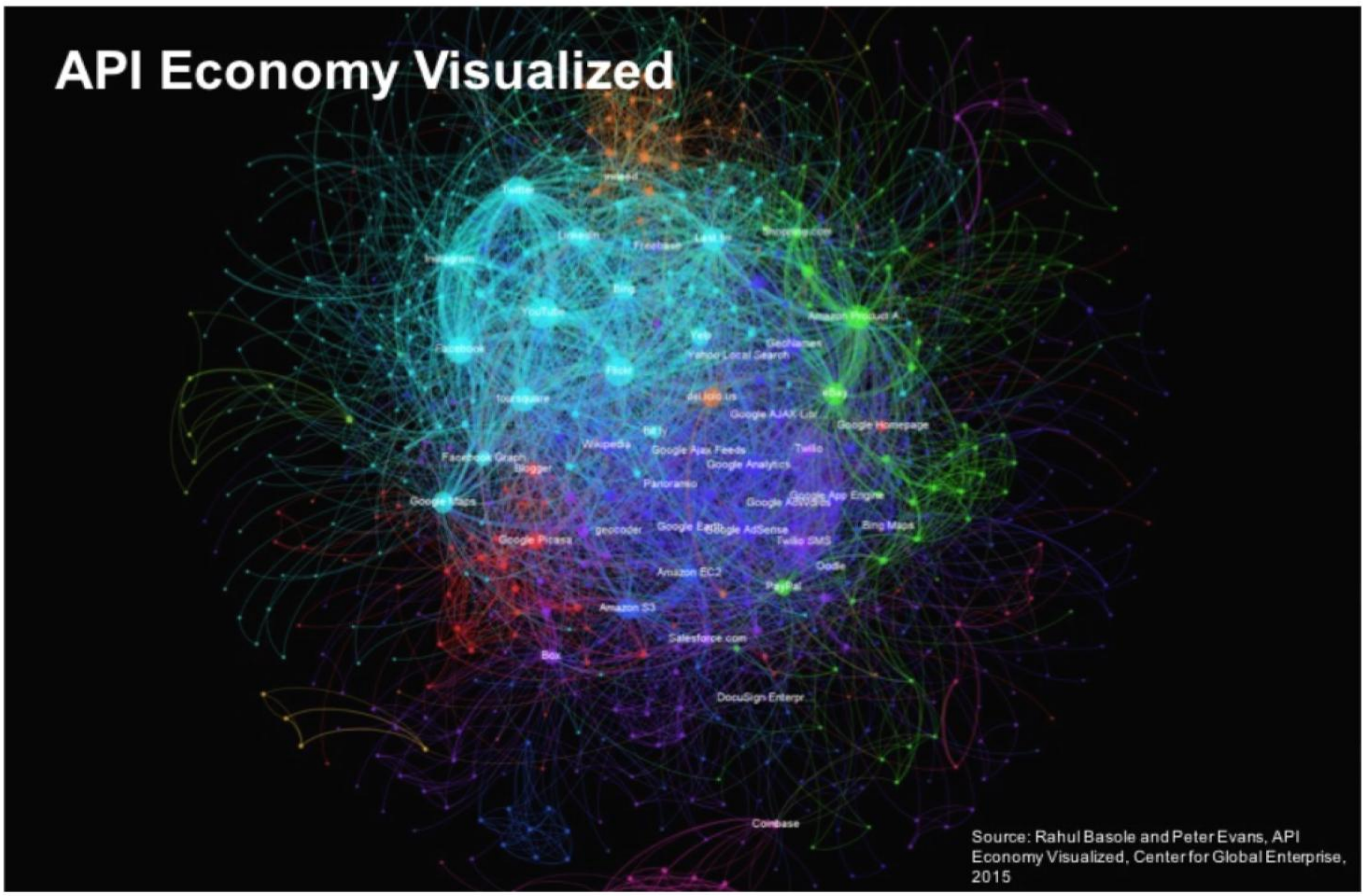
How well are the people aligned with the strategic direction of the organization? Can people in your organization clearly state the value proposition of your business? Do your customers understand it and experience that value? Customer digital behavior can be represented in customer journey maps. Has your organization taken the time to understand these journeys and what experiences potential and current customers face?

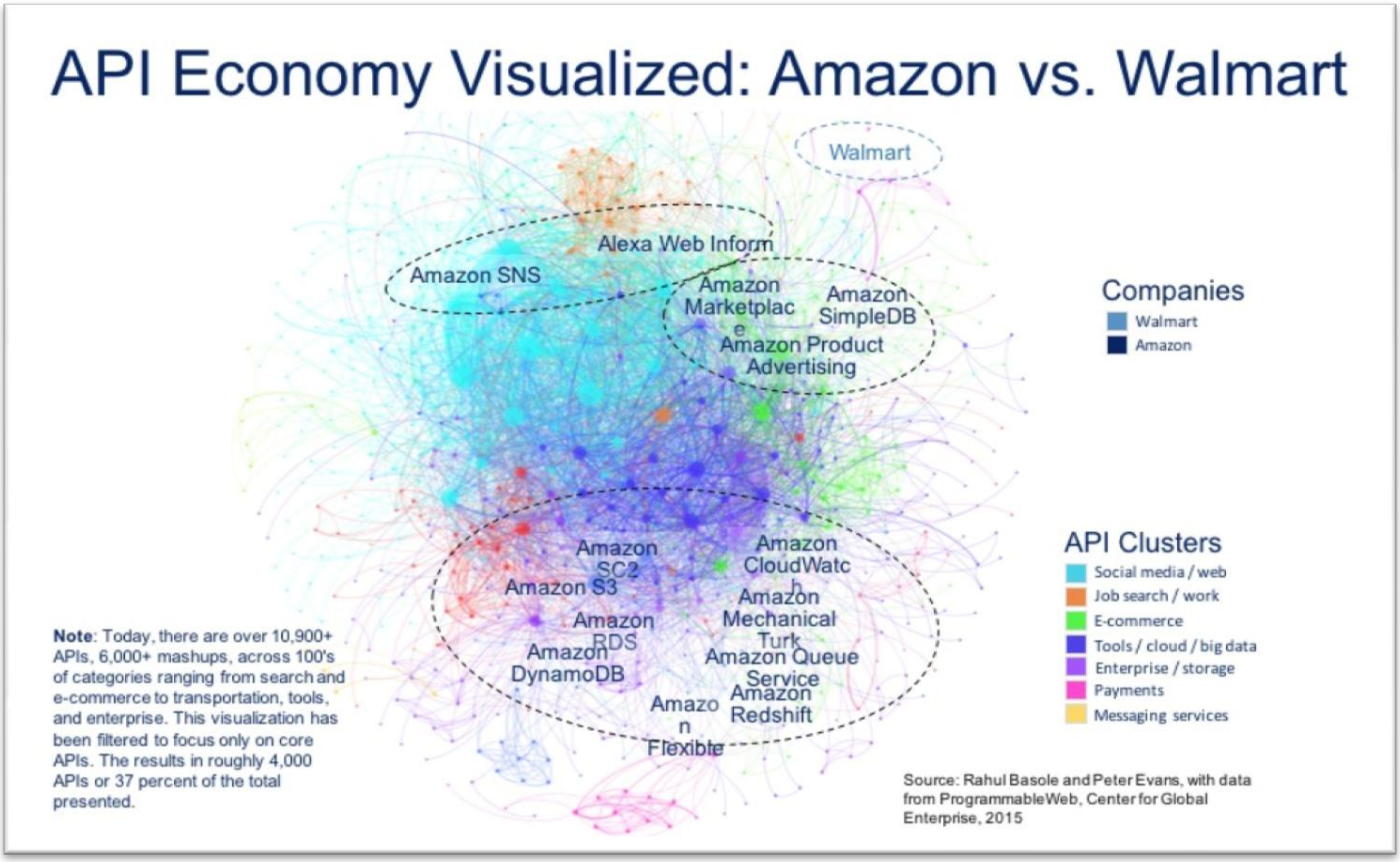
Understand the device mesh in your industry and how those devices provide the desired ambient experience for your customers. Does your systems architecture capture your client's journey in the sphere of Information of Everything? Companies need to begin thinking of their Mesh App and Service Architecture model and their Internet of Things Platforms. This is a strategic perspective with will take time and develop over a long span.

Once the above questions are answered ensure that the business model and related capabilities are in place. Where there are gaps create an execution model to fill in the space.

This article highlights the need to understand and be proactive. It outlines a set of activities and questions to address the digital mesh facing your industry. This article does not go into the deal to solve your digital operating model, but it lays the foundation in place from which to build. The best way to create a successful future is to be an influencer and make the changes that benefit your organization.

Here is a final idea to think about. Look at the image on the following page that shows the global flight paths (Cheshire 2013). It is interesting to compare and contrast that to the API ecosystem image. Why are the nodes located where they are? What does it mean to your business to be on a node? What happens to a business in a “fly over state”? Is the API ecosystem being set for the next business generation just as previous generations built our flight infrastructure?




API Economy Visualized: Amazon vs. Walmart
Walmart
Alexa Web Inform
Amazon SNS
Amazon Marketplac e
Amazon SimpleDB
Amazon Product Advertising
Companies
Walmart
Amazon
Amazon SC2
Amazon S3
Amazon RDS
Amazon DynamoDB
Amazon CloudWatc h
Amazon Mechanical Turk
Amazon Queue Service
Amazon Redshift
Amazo n Flexible
API Clusters
Social media / web
Job search / work
E-commerce
Tools / cloud / big data
Enterprise / storage
Payments
Messaging services
Note: Today, there are over 10,900+ APIs, 6,000+ mashups, across 100's of categories ranging from search and e-commerce to transportation, tools, and enterprise. This visualization has been filtered to focus only on core APIs. The results in roughly 4,000 APIs or 37 percent of the total presented.
Source: Rahul Basole and Peter Evans, with data from ProgrammableWeb, Center for Global Enterprise, 2015


Airline*
Emirates
Lufthansa
Delta Air Lines
Air France
British Airways
Cathay Pacific
Singapore Airlines
*Ordered by scheduled international passenger kilometres flown in 2010 (source Wikipedia).
Only routes in the OpenFlights database are plotted.
Map: James Cheshire, spatialanalysis.co.uk
Flights Data: openflights.org
Basemap Data: naturalearthdata.com

ABOUT THE AUTHOR

Michael Hinnebusch is an innovative business strategist and a consultant with TCS's Global Consulting Practice focusing on Business Strategy & Design. He leverages 20+ years of industry and consulting experience across multiple industries. Mike specializes in helping organizations translate their business strategy into actionable programs to deliver value to the organization and their clients.

Mike holds a Bachelor of Science degree from the University of Pittsburgh in Mechanical Engineering. He received his Master's in Business Administration from the University of Texas, El Paso. Michael currently resides in Ohio, USA.

REFERENCES

Gartner, October 6, 2015; refer to: www.gartner.com/newsroom/id/3143521 (accessed February 23, 2016).

S. Galer: Gartner's Top 10 Strategic Technology Trends For 2016, December 21, 2015; refer to: www.digitalistmag.com/digital-economy/2015/12/21/gartners-top-10-strategic-technology-trends-2016-03857991 (accessed February 23, 2016).

R.C. Basole, P.C. Evans: Decoding the API Economy with Visual Analytics, September 2, 2015; refer to: https://thecge.net/decoding-the-api-economy-with-visual-analytics/ (accessed February 2, 2016). Image used by permission of the author.

The Center for Global Enterprise: About the CGE, 2016; refer to: https://thecge.net/about/ (accessed February 24, 2016).

J. Cheshire: What's so Great About a World Flight Paths Map?, May 30, 2013; refer to: http://spatial.ly/2013/05/great-world-flight-paths-map/ (accessed February 29, 2016). Image used by permission of the author.

Article

Addressing Enterprise Change Capability, a Constraint in Business Transformation

Inji Wijgunaratne and Sharma Madiraju

Abstract

Evidence shows that, more often than not, large IT programs do not succeed, exceeding their budgets, timelines, and delivering abbreviated scope and value. This article endeavors to observe and assess the problem from a capability perspective. We argue that though the solution or the future state is often focused upon and specified, the same level of attention is not devoted to the capabilities – both business and IT – required to bring about the organizational transition. Since the level and scale of transformational capabilities are very different from those needed to run a "business as usual" operation, this mismatch is at the heart of the problem. We then discuss a relatively inexpensive approach to remedy the issue.

Keywords

Enterprise Architecture, Business Transformation, Program Management, Business Capabilities

INTRODUCTION

More often than not, large transform programs have failed – either failed to deliver their original outcomes, blown their budgets and/or timeframes, in many instances, all of the above. A research study undertaken in 2012 on more than 5,400 IT projects by the consulting firm McKinsey and the BT Centre for Major Programme Management at the University of Oxford (McKinsey 2012) found that half of all large IT projects (defined as those with an initial budget in excess of $(USD)15 million) significantly exceeded their budgets; on average these projects ran 45% over budget and delivered 56% less value than originally scoped. These IT projects had a combined cost overrun of a massive $(USD)66 billion. Sometimes large IT projects perform so badly that they threatened the very existence of the company: 17% of large IT projects (with cost overruns of more than 200%) fell into this category. Closer to where the authors reside, The AGE, a leading Australian newspaper, published an article in 2014 (Sexton 2014) referencing a report by the Ombudsman for the State of Victoria on the performance of ten large IT programs in the public sector. The report found that the combined cost of delivering them was $(AUD)1.44 billion more than originally budgeted.

The evidence is damning, but certainly accords with anecdotal observations from the authors' own experience and that of several of their colleagues. In our experience, it is very seldom that large IT programs actually ticked all the boxes of success.

Why is this so? Surely companies do not rush into these large projects ill-prepared? Conventional wisdom suggests that companies *do* prepare; however, unexpected and unforeseen confluences of circumstance thwart and ultimately derail the progress of these endeavors.

How can the discipline of enterprise architecture assist with their preparation and execution? Similarly, the traditional or customary view is that a mature and effective enterprise architecture process with a clear line of sight between the enterprise, program, and solution architectures, with appropriate mechanisms for architecture governance helps the efficient implementation of large IS programs of work.

These reasons may be valid; however, in this article we present another thread of reasoning that, we believe, goes to the heart of the matter.

AN EXAMPLE SCENARIO

Imagine an enterprise, Acme Widgets and Insurance Ltd. They typically undertake IT projects including business programs with a major IT component. They include "new" projects plus run-of-the-mill maintenance and enhancement of existing systems. These IT projects vary in size from a few hundred thousand to millions of dollars, with a project in the scale of $10 million being considered a "large" project. The organization is generally comfortable delivering projects within this range, having gained experience handling them through repeated delivery.

Now comes a business transformation initiative estimated at $100 million, 10 times larger than their conventional “large” project. What are the chances that Acme Widgets and Insurance would succeed?

“Not much”, many would say. “The scale of the initiative will take Acme Widgets and Insurance out of their comfort zone on several fronts.”

This assessment, though informal, goes to the crux of the issue: capabilities. The capabilities that underpin the success in the sub-$10M range of IS-driven organizational change is vastly different from those needed for a large business transformational endeavor.

- Qualitatively: they will need capabilities that currently they do not possess.
- Quantitatively: the scale and maturity levels of certain capabilities they do possess need to increase significantly.

For example, the business transformation initiative will entail a “program of work” needing the execution of several simultaneous workstreams, each with its own projects – larger teams, complex dependencies and impacts, engendering deeper and broader organizational change – requiring program and change management capabilities that Acme Widgets and Insurance currently do not possess. Moreover, the scale of the initiative points to vendor partners’ assistance as Acme simply cannot scale up their in-house resources to the required levels quickly – and Acme has historically managed a few second-tier vendor/supplier organizations – needing vendor management capabilities of a nature hitherto unfamiliar at Acme. Moreover, Acme will need several non-production environments to support the program, and associated release and configuration management skills – out of Acme’s current league; sustained build, test, and delivery of releases over three years at a throughput well out of Acme’s comfort zone; and the list goes on.

The rest of this article explores these ideas in a more formal fashion.

CHANGE CAPABILITY AS A CONSTRAINT IN BUSINESS TRANSFORMATION

The key focus of a change initiative is the future state of the enterprise. Typically, changes associated with a smaller initiative are limited, focusing on a contained area within the business and involving contained change to the existing IT landscape; sometimes, for example where operational IT efficiencies are sought, some IT-driven projects should only impact the IT environment with no perceptible changes to business process. A large initiative, though, typically aims to achieve strategic business objectives and realize a comprehensive future state involving many aspects of the business enterprise; in other words to introduce significant change to the business or operating model of the enterprise. Consequently, the IT landscape will be earmarked for significant and complex change, either in terms of large-scale development of new systems, rationalization of existing systems, or both. Whereas a small initiative is likely to spawn a few smaller projects to deliver a single solution, the realization of strategic business outcomes through a large transformational initiative involves a program of work consisting of several related major projects.

For a large initiative, a significant effort typically goes into describing the future state, addressing the business operating model, business capabilities, business processes and relevant technology architectures, and solutions that support them. The program methodology and artifacts will describe the projects, their deliverables, and their sequence depicting the transition from current to future states (items1, 2, and 3 in Figure 1; figures are located at the end of article). A clear line of sight between the various artifacts of this exercise, their creation, maintenance, and governance, are factors quite correctly identified as being important requisites for the success of endeavors of this nature.

However, a glance at Figure 1 indicates that these measures are certainly necessary, but they alone are not enough to bring about success. As we commented above, a large-scale transformation involves very significant change on many fronts – to the business operating model, organizational structure and culture, to the IT systems landscape. The “elephant in the room” is the organization’s ability to deliver this change successfully. Item 4 in Figure 1 displays this requirement, making clear that the change that achieves the transformation must be actively driven, by the team that drives the transformation (shown here as the Business and IT change agents).

While the new business operating model hopefully delivers the intended strategic business outcomes, it is also critically important that the enterprise actually makes the transition; that it is successfully reshaped, changed, evolved. Accordingly, the critical capabilities (both business and IT capabilities) that are required to deliver change – their current maturity, scale, tooling, etc. – must be subject to a level of scrutiny as stringent as that for the future state. In the authors’ experience, however, these critical capabilities are not subjected to a level of scrutiny anywhere near that exercised for defining the desired solution or future enterprise.

In the pages that follow, we explore how we can leverage the enterprise architecture discipline and artifacts to address transformational capabilities in more detail.

BUSINESS CAPABILITY MAPS IN ENTERPRISE ARCHITECTURE

A key discipline of enterprise architecture is business architecture – and it is the artifacts of business architecture that are first brought into existence to describe the current and/or desired state of an enterprise or a business domain.

Figure 2 depicts the representational technique of enterprise architecture known as a business capability map.

Business capability maps, being a fundamental artifact of business architecture, have many uses; one such is its role in assessing the alignment with business objectives. Validating the business capability map of the enterprise (or business domain) against the strategic objectives of the business highlights:

- The business capabilities that are important to the strategic imperatives of the business
- The degree to which these capabilities are currently able to support these strategic business objectives (due to current maturity, focus, resource, etc. constraints of these capabilities)

This type of artifact, where each business capability is color-coded – say into one of four colors depicting its relative value/contribution to a business strategy – is termed a "heat map". Using this technique we can develop a heat map depicting the importance of business capabilities to the strategic business objectives; and another depicting the capability maturity of the business capabilities.

With the assistance of these "heat maps", we are able to determine how to uplift, scale, or otherwise evolve a capability to a required level to appropriately support the enterprise's business objectives.

Can we employ business capability maps in a similar fashion, to assess the organization's capability to effect and deliver change? Let us explore this question further.

BUSINESS CHANGE AGENTS

Returning to Figure 1, we notice that there are two sets of change agents; those responsible for Business and IT change. Let us first examine business change agency.

We return to the company of the sample scenario, Acme Widgets and Insurance.

The business capability map for Acme Widgets and Insurance is shown in Figure 3.

This business capability map has been used extensively in describing the future state for Acme. A heat map has been derived from this, depicting the current misalignment of business capability with the strategic objectives of the business. This heat map has been used as a fundamental building block in describing the target state business outcomes and in expressing the journey from current to target – by refocusing certain business capabilities, investing in uplifting the maturity of certain capabilities, and moving the focus away from some others.

Figure 3 represents the base capability map *sans* the heat mapping for the new solution; Figure 4 shows the strategic capabilities that need attention for this business transformation program. Also displayed in Figure 4 are the alignment gaps for these key capabilities (red = high misalignment; orange = medium misalignment).

The point of this thread of reasoning though is different: a review of the business capability map of Figure 3 reveals a curious fact. The map describes a steady state insurance business – capabilities to plan, govern, operate, and support the insurance business are clearly represented.

But where are the capabilities for driving business change, making changes to the way the insurance business is run? At first glance, "New business planning and review" in Sales and "Organizational Development" in HR are the only capabilities that show promise. On closer investigation (through talking to the business stakeholders) though, the former is revealed to be business development – the sales capability of expanding the existing business by finding and managing new clients; the latter is the HR capability that assesses new skills required in the enterprise and organizes training and other support to facilitate skills development. This latter capability is able to support the *status quo* by developing skills to maintain the existing operating model efficiently. It can also function to support the change to a new operating model that is to be a support function for a change agent. However, this capability is not the change agency itself. It transpires therefore that no business "change agent" capabilities are depicted in Acme's business capability map. We are faced with a dilemma: on one hand, it is abundantly clear that in order to realize the desired target state of the business you need to change the way the business is run. On the other hand, within the enterprise, the business capabilities to effect these changes are virtually non-existent.

Acme Widgets and Insurance illustrates a dichotomy that is often manifest in enterprises. Business domains possess capabilities to perform existing business functions – plan, organize, implement, monitor; in other words they typically reflect the needs of a steady state.

The same level of focus is not given to capabilities needed to effect change to an operating model. Business change agent capabilities such as organizational design, business process re-engineering, organizational change management, are required but rarely called out explicitly.

In essence, mature change agent capabilities with experience in managing complex change are required to deliver organizational change successfully. Typically though, these capabilities either do not exist in organizations, or where they do exist, do not possess the depth of skills or the scale to effect large-scale organizational change.

IT CHANGE AGENTS

Let us now turn our attention to Acme's IT division.

The "business" capability map for Acme's IT division is shown in Figure 5. This is an expansion of the row of IT capabilities under "enabling" business capabilities.

At first glance, from the perspective of managing change, IT capabilities appear more promising. An IT division deals with change to a far greater extent than business operations, and these capabilities are reflected in IT's capability map.

Let us take a closer look. An analysis of the IT capability map reveals the following critical capabilities for transformation; i.e., for planning, designing, and implementing large-scale change.

- Architecture – enterprise architecture (program architecture), solution architecture
- IT Governance
- Customer Management
- Portfolio and Project Management
- Service Management – service transition, service operation
- Shared Services Management – infrastructure management
- Technology Build
- Demand Management
- Vendor Management

Figures 6 and 7 show the partial results of a heat mapping exercise, similar to that described in the section "Business Change Agents", but with a focus on the IT capabilities needed to support a large transformation. Figure 6 presents five of the nine capabilities deemed to be critically important for transformation, with a gap analysis between existing and desired levels of capability. Figure 7 transposes the gap levels against the IT capability map. This heat mapping process clearly indicates the gaps in capability to support a large-scale transformation.

DISCUSSION

The more formal analysis we have undertaken bears out the informal deductions made earlier in the section "Change Capability as a Constraint in Business Architecture"**Error! Reference source not found.**.

The analysis shows the underlying problem with capabilities to successfully handle large transformation programs, both from a business and an IT perspective. IT change capabilities are discernible in a typical IT department and consequently would appear in a capability map describing its operations. However, their level of maturity is rarely sufficient to drive large-scale change. On the other hand, as shown earlier, business capabilities to drive change rarely even appear in a business capability map – and are rarely evident in the operations of an enterprise

This in itself is not surprising. To hark back to the example presented in the section "An Example Scenario"**Error! Reference source not found.**, usiness executives will argue that it makes no business sense for organizations to continuously invest in developing and maintaining capability levels to support large transformational change, which may only occur once in several years. Indeed, they will argue that their organization only needs to manage incremental changes on a routine basis, and that they have invested to successfully manage their day-to-day change requirements.

On one hand, the above argument has merit – organizations need not invest in developing and maintaining transformational change agent capabilities as if they need to embark on a large-scale transformation every day. On the other hand, today's catch-call is business agility, nimbleness, disruption, innovation (McKinsey 2015a, b). It is very easy to step back and observe the plethora of technology-driven changes that business environments have experienced since the dawn of this millennium. Businesses that focus only on capabilities to maintain the *status quo* will invariably encounter difficulties and eventually will decline due to the tides of business and technology change.

Accordingly, the truth is somewhere in the middle.

Organizations should explore cost-effective and nimble ways to address the capability needs of supporting large-scale business and technological changes. As a minimum, organizations need to understand *what* capabilities they possess and also design *how* to uplift these capabilities to handle large-scale change. In other words, the appropriate middle path is to invest and develop *awareness* and *readiness* for transformational change.

- *Awareness*: invest in undertaking a self-examination to determine the critical capabilities and their maturity levels needed to support different scales of business transformation. This can be achieved through the capability heat map development we have illustrated for IT capabilities. The same technique can be employed in developing a business capability heat map for managing change.
- *Readiness*: Develop contingency plans for capability uplift to different target levels of change capability. Again, contingency planning can be undertaken both for business and IT capability management. This may involve tactics such as tooling, in-house skills uplift, employing a selected partner to scale up on demand, etc. It may also involve scrutiny and conscious paring down of non-value-adding complexity (both business and IT complexity) that companies tend to accumulate (BCG 2013) over the years.

This process may be revisited at regular intervals, to keep the contingency plan current. In this way, when the actual need becomes visible in the horizon – large transformation initiatives do take time between conception and implementation – the appropriate change agent capability uplift plan can be put in motion.

We believe that this pragmatic, relatively inexpensive approach will dramatically increase the chances of success of large transformation endeavors.

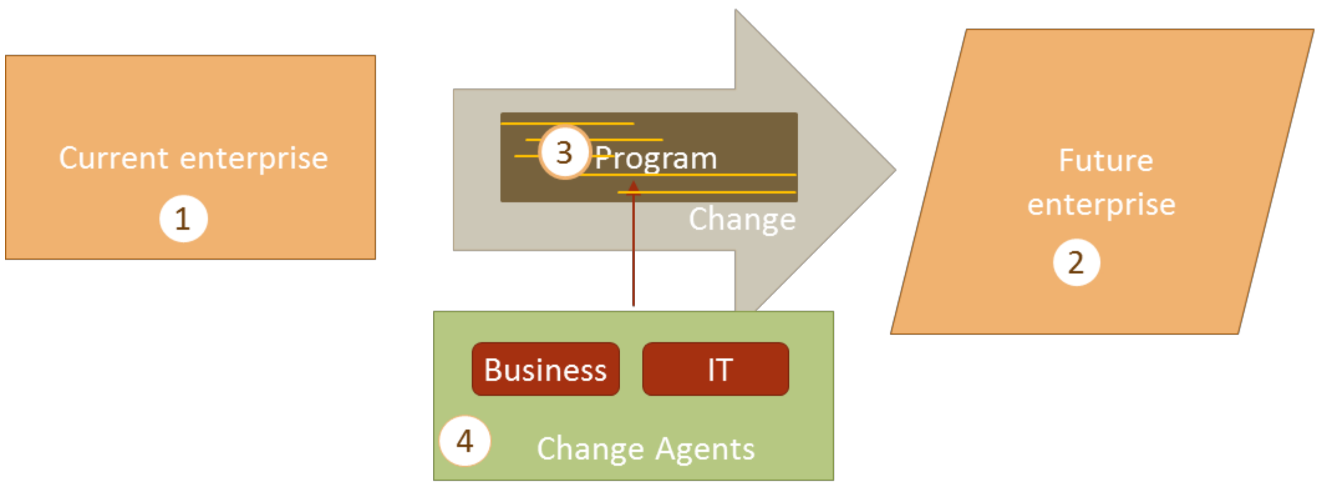


Figure 1

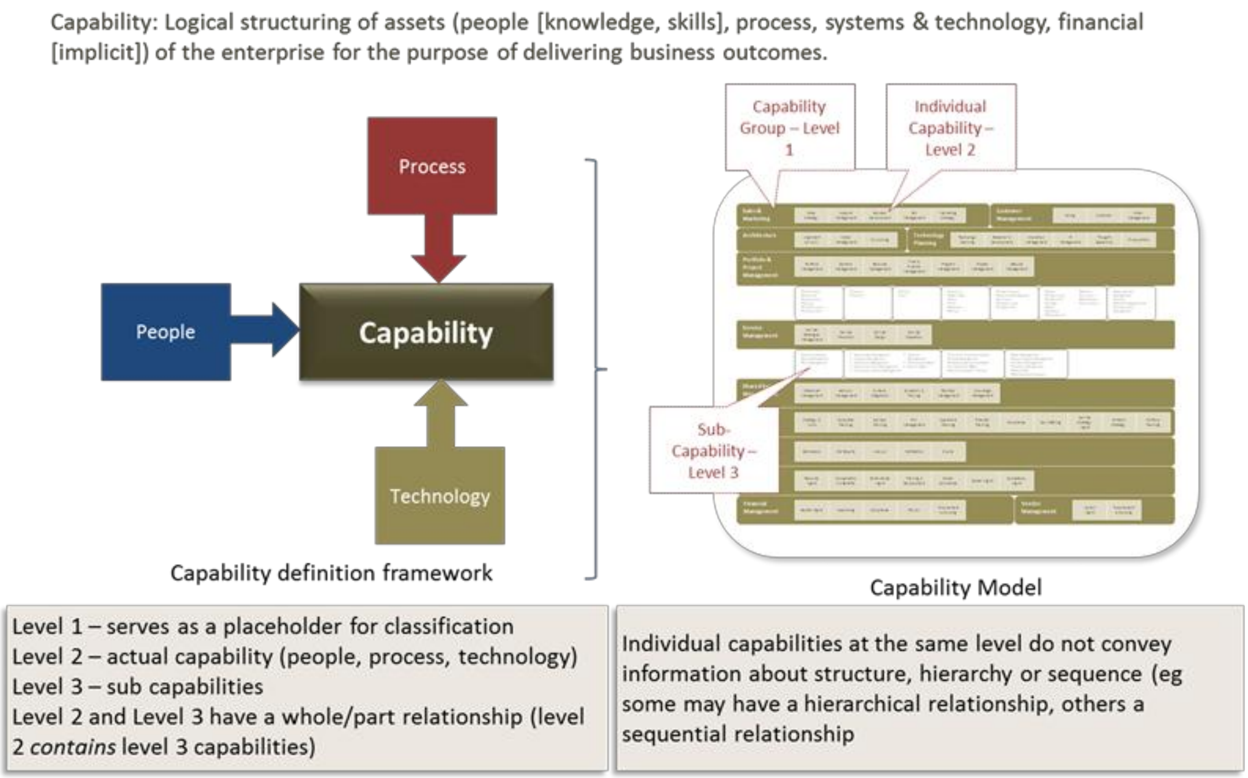

Capability: Logical structuring of assets (people [knowledge, skills], process, systems & technology, financial [implicit]) of the enterprise for the purpose of delivering business outcomes.
Process
People
Capability
Technology
Capability definition framework
Capability Group – Level 1
Individual Capability – Level 2
Sub-Capability – Level 3
Capability Model
Level 1 – serves as a placeholder for classification
Level 2 – actual capability (people, process, technology)
Level 3 – sub capabilities
Level 2 and Level 3 have a whole/part relationship (level 2 contains level 3 capabilities)
Individual capabilities at the same level do not convey information about structure, hierarchy or sequence (eg some may have a hierarchical relationship, others a sequential relationship

Figure 2

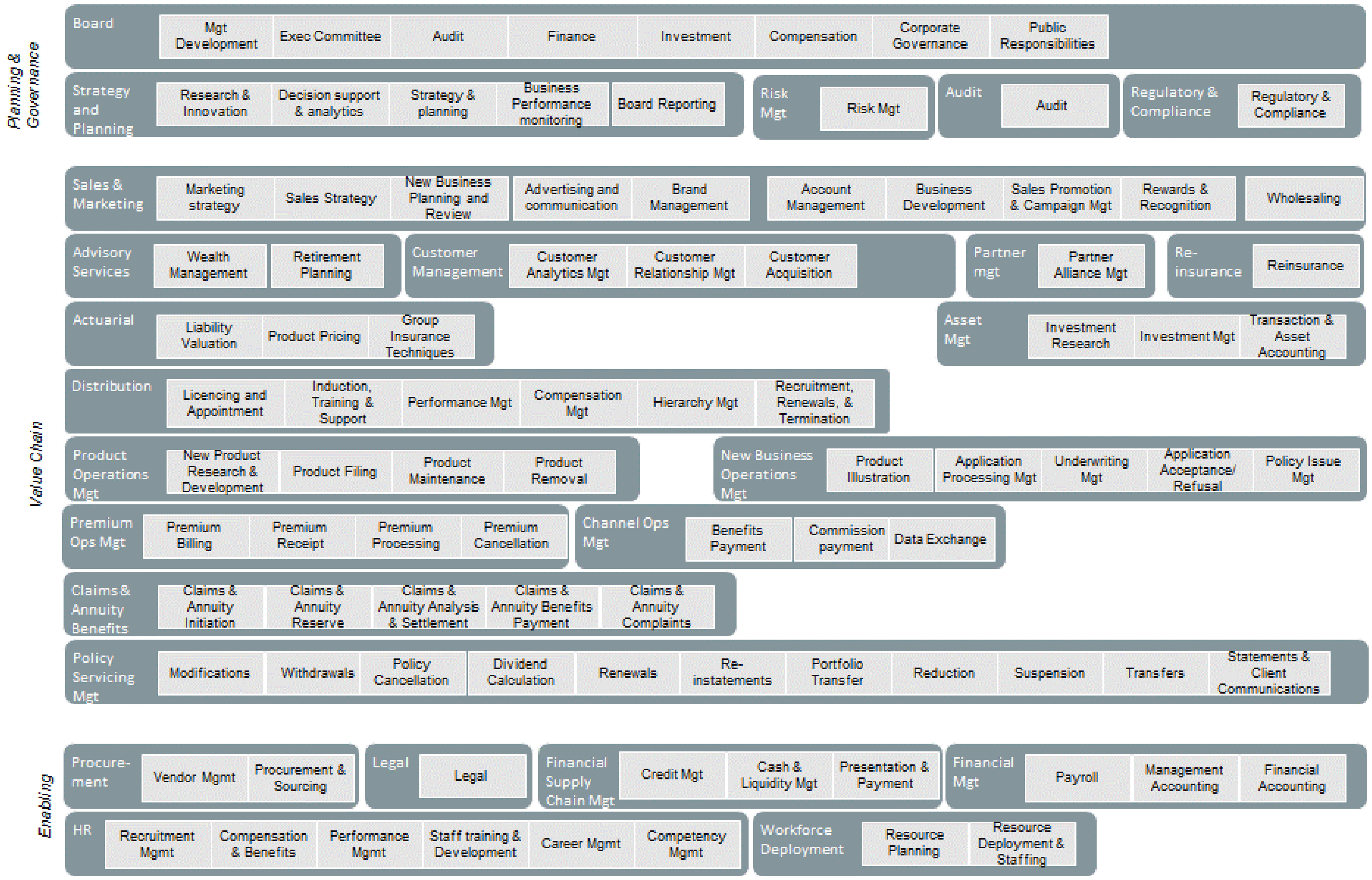

Planning & Governance
Board
Mgt Development
Exec Committee
Audit
Finance
Investment
Compensation
Corporate Governance
Public Responsibilities
Strategy and Planning
Research & Innovation
Decision support & analytics
Strategy & planning
Business Performance monitoring
Board Reporting
Risk Mgt
Risk Mgt
Audit
Audit
Regulatory & Compliance
Regulatory & Compliance
Value Chain
Sales & Marketing
Marketing strategy
Sales Strategy
New Business Planning and Review
Advertising and communication
Brand Management
Account Management
Business Development
Sales Promotion & Campaign Mgt
Rewards & Recognition
Wholesaling
Advisory Services
Wealth Management
Retirement Planning
Customer Management
Customer Analytics Mgt
Customer Relationship Mgt
Customer Acquisition
Partner mgt
Partner Alliance Mgt
Re-insurance
Reinsurance
Actuarial
Liability Valuation
Product Pricing
Group Insurance Techniques
Asset Mgt
Investment Research
Investment Mgt
Transaction & Asset Accounting
Distribution
Licencing and Appointment
Induction, Training & Support
Performance Mgt
Compensation Mgt
Hierarchy Mgt
Recruitment, Renewals, & Termination
Product Operations Mgt
New Product Research & Development
Product Filing
Product Maintenance
Product Removal
New Business Operations Mgt
Product Illustration
Application Processing Mgt
Underwriting Mgt
Application Acceptance/ Refusal
Policy Issue Mgt
Premium Ops Mgt
Premium Billing
Premium Receipt
Premium Processing
Premium Cancellation
Channel Ops Mgt
Benefits Payment
Commission payment
Data Exchange
Claims & Annuity Benefits
Claims & Annuity Initiation
Claims & Annuity Reserve
Claims & Annuity Analysis & Settlement
Claims & Annuity Benefits Payment
Claims & Annuity Complaints
Policy Servicing Mgt
Modifications
Withdrawals
Policy Cancellation
Dividend Calculation
Renewals
Re-instatements
Portfolio Transfer
Reduction
Suspension
Transfers
Statements & Client Communications
Enabling
Procure-ment
Vendor Mgmt
Procurement & Sourcing
Legal
Legal
Financial Supply Chain Mgt
Credit Mgt
Cash & Liquidity Mgt
Presentation & Payment
Financial Mgt
Payroll
Management Accounting
Financial Accounting
HR
Recruitment Mgmt
Compensation & Benefits
Performance Mgmt
Staff training & Development
Career Mgmt
Competency Mgmt
Workforce Deployment
Resource Planning
Resource Deployment & Staffing

Figure 3

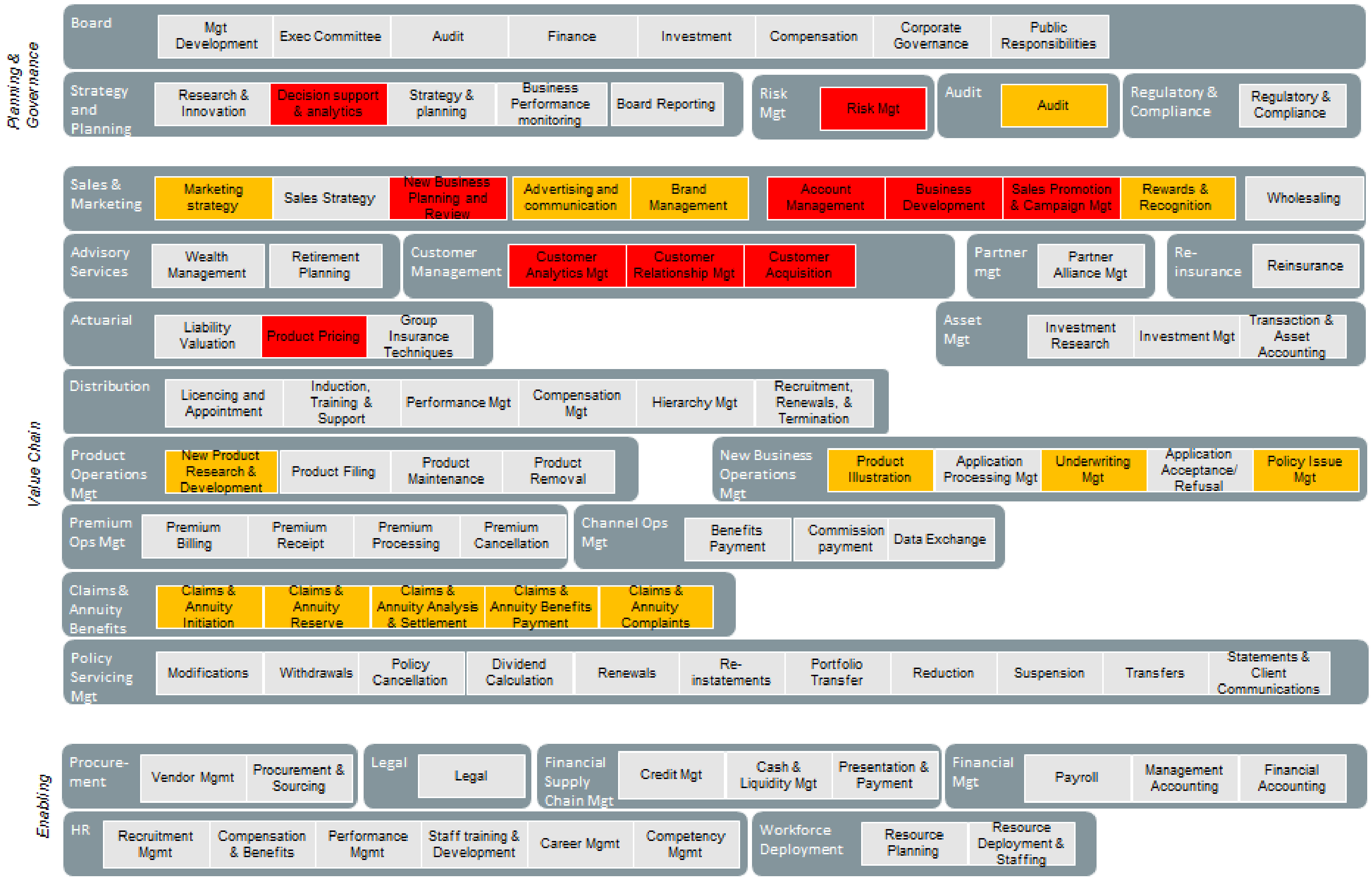
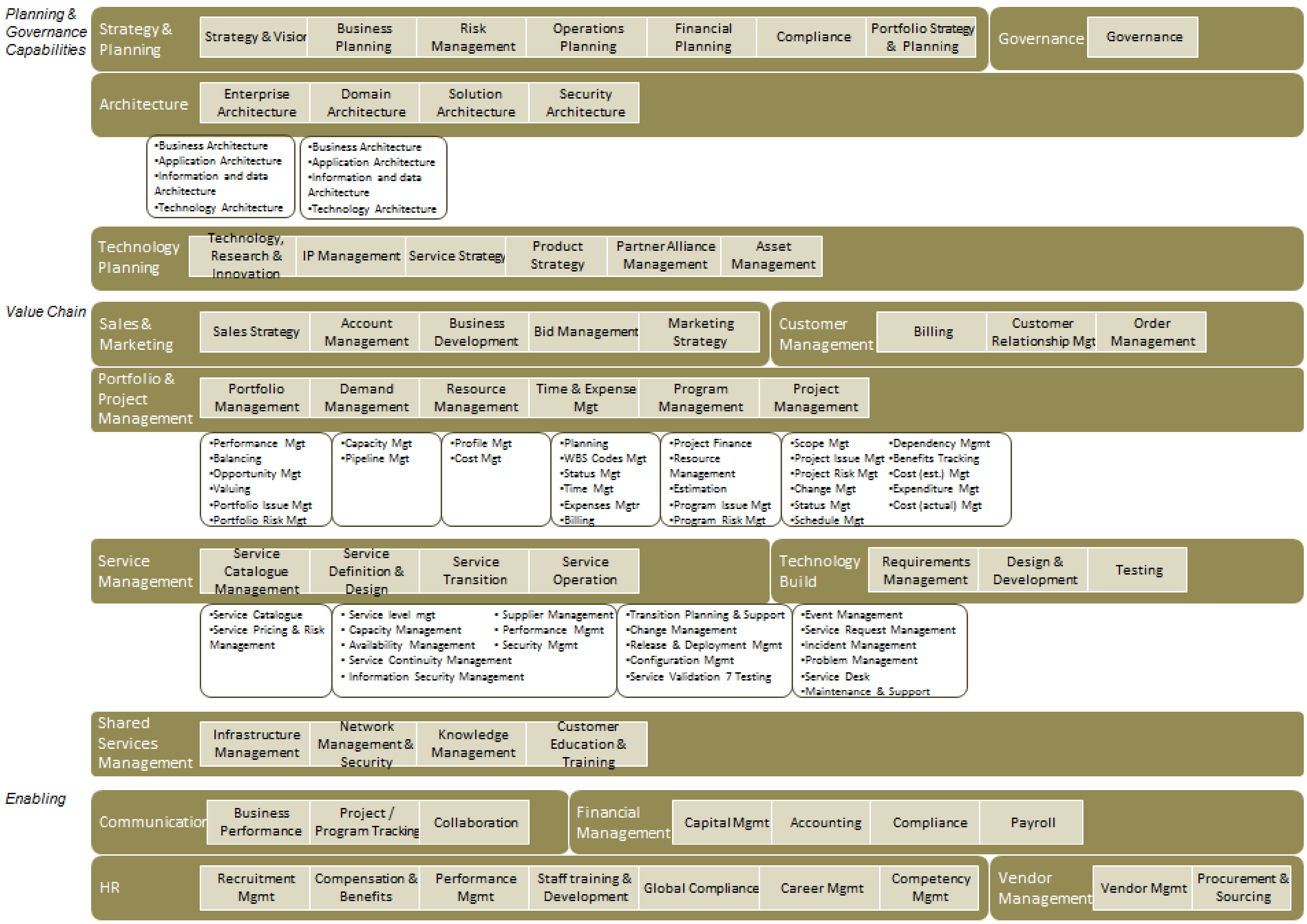

Planning & Governance
Board
Mgt Development
Exec Committee
Audit
Finance
Investment
Compensation
Corporate Governance
Public Responsibilities
Strategy and Planning
Research & Innovation
Decision support & analytics
Strategy & planning
Business Performance monitoring
Board Reporting
Risk Mgt
Risk Mgt
Audit
Audit
Regulatory & Compliance
Regulatory & Compliance
Value Chain
Sales & Marketing
Marketing strategy
Sales Strategy
New Business Planning and Review
Advertising and communication
Brand Management
Account Management
Business Development
Sales Promotion & Campaign Mgt
Rewards & Recognition
Wholesaling
Advisory Services
Wealth Management
Retirement Planning
Customer Management
Customer Analytics Mgt
Customer Relationship Mgt
Customer Acquisition
Partner mgt
Partner Alliance Mgt
Re-insurance
Reinsurance
Actuarial
Liability Valuation
Product Pricing
Group Insurance Techniques
Asset Mgt
Investment Research
Investment Mgt
Transaction & Asset Accounting
Distribution
Licencing and Appointment
Induction, Training & Support
Performance Mgt
Compensation Mgt
Hierarchy Mgt
Recruitment, Renewals, & Termination
Product Operations Mgt
New Product Research & Development
Product Filing
Product Maintenance
Product Removal
New Business Operations Mgt
Product Illustration
Application Processing Mgt
Underwriting Mgt
Application Acceptance/ Refusal
Policy Issue Mgt
Premium Ops Mgt
Premium Billing
Premium Receipt
Premium Processing
Premium Cancellation
Channel Ops Mgt
Benefits Payment
Commission payment
Data Exchange
Claims & Annuity Benefits
Claims & Annuity Initiation
Claims & Annuity Reserve
Claims & Annuity Analysis & Settlement
Claims & Annuity Benefits Payment
Claims & Annuity Complaints
Policy Servicing Mgt
Modifications
Withdrawals
Policy Cancellation
Dividend Calculation
Renewals
Re-instatements
Portfolio Transfer
Reduction
Suspension
Transfers
Statements & Client Communications
Enabling
Procure-ment
Vendor Mgmt
Procurement & Sourcing
Legal
Legal
Financial Supply Chain Mgt
Credit Mgt
Cash & Liquidity Mgt
Presentation & Payment
Financial Mgt
Payroll
Management Accounting
Financial Accounting
HR
Recruitment Mgmt
Compensation & Benefits
Performance Mgmt
Staff training & Development
Career Mgmt
Competency Mgmt
Workforce Deployment
Resource Planning
Resource Deployment & Staffing

Figure 4



Planning & Governance Capabilities
Strategy & Planning
Strategy & Vision
Business Planning
Risk Management
Operations Planning
Financial Planning
Compliance
Portfolio Strategy & Planning
Governance
Governance
Architecture
Enterprise Architecture
Domain Architecture
Solution Architecture
Security Architecture
Technology Planning
Technology, Research & Innovation
IP Management
Service Strategy
Product Strategy
Partner Alliance Management
Asset Management
Value Chain
Sales & Marketing
Sales Strategy
Account Management
Business Development
Bid Management
Marketing Strategy
Customer Management
Billing
Customer Relationship Mgt
Order Management
Portfolio & Project Management
Portfolio Management
Demand Management
Resource Management
Time & Expense Mgt
Program Management
Project Management
Service Management
Service Catalogue Management
Service Definition & Design
Service Transition
Service Operation
Technology Build
Requirements Management
Design & Development
Testing
Shared Services Management
Infrastructure Management
Network Management & Security
Knowledge Management
Customer Education & Training
Enabling
Communication
Business Performance
Project / Program Tracking
Collaboration
Financial Management
Capital Mgmt
Accounting
Compliance
Payroll
HR
Recruitment Mgmt
Compensation & Benefits
Performance Mgmt
Staff training & Development
Global Compliance
Career Mgmt
Competency Mgmt
Vendor Management
Vendor Mgmt
Procurement & Sourcing

Figure 5

Capability	Desired Capability Level	Current Capability Level	Criticality for Transformation	Gap
Architecture: EA (Program Architecture)	• Ability to identify and agree key business imperatives, outcomes; create business, application, technology architectures at business domain/enterprise level to realize the objectives • Ability to express a transformation program in EA terms, tying outcomes to business, application, technology architecture pieces; obtaining business exec buy-in • Ability to manage the EA model as it evolves	• Implemented "IS/IT"-centric EA models. None have been visible to the business stakeholders. • No experience in tying business strategy and EA, in representing a program of work in terms of business, application, tech architectures. • No tool to manage EA/program architecture artifacts.	High	High
Architecture: Solution Architecture	• Ability to develop and express a solution comprising business, application, tech architecture components that realize a defined set of requirements; document key decisions, identify risks • Ability to trace solution components to strategic outcomes/benefits of the business	• Capability exists for medium to low complexity solutions. • Hitherto have not traced solution components to business strategic outcomes.	High	Med
IS Customer Management	• Understand (and anticipate) express and implicit customer needs, manage customer expectations, maintain relationships with key stakeholders, ability to shape delivery to ensure best customer satisfaction outcomes • Ability to successfully collaborate with the business in navigating large, complex, inter-related work streams delivering business outcomes	• Some customer management experience, • No previous experience with collaborating with the business and managing business stakeholders, in large, complex, programs of work aimed at engendering major organizational change.	High	Med
Portfolio and Project Management: Program Management	• Ability to develop a WBS and accurately estimate the work efforts for large, related sets of work streams comprising multiple individual projects • Ability to report on progress of complex, inter-related program of work in an insightful and accurate manner • The leadership to manage a complex, inter-related program of work involving in-house and multi-vendor teams	• Project managers with experience of projects of <$10M total spend. • No in-house program management leadership skills. • Portfolio and program reporting process exists, but not tested beyond the current bandwidth of BAU and business projects.	High	**High**
Portfolio and Project Management: Benefits Management	• Ability to tie business benefits to solution components at a granular level • Experience in assessing variations against benefit impact • Experience in tracking benefit realization once a system is in operation	• Currently no benefits management experience (business cases are created at project initiation but not tracked thereafter).	High	**High**
. . .	. . .	. . .		

Figure 6

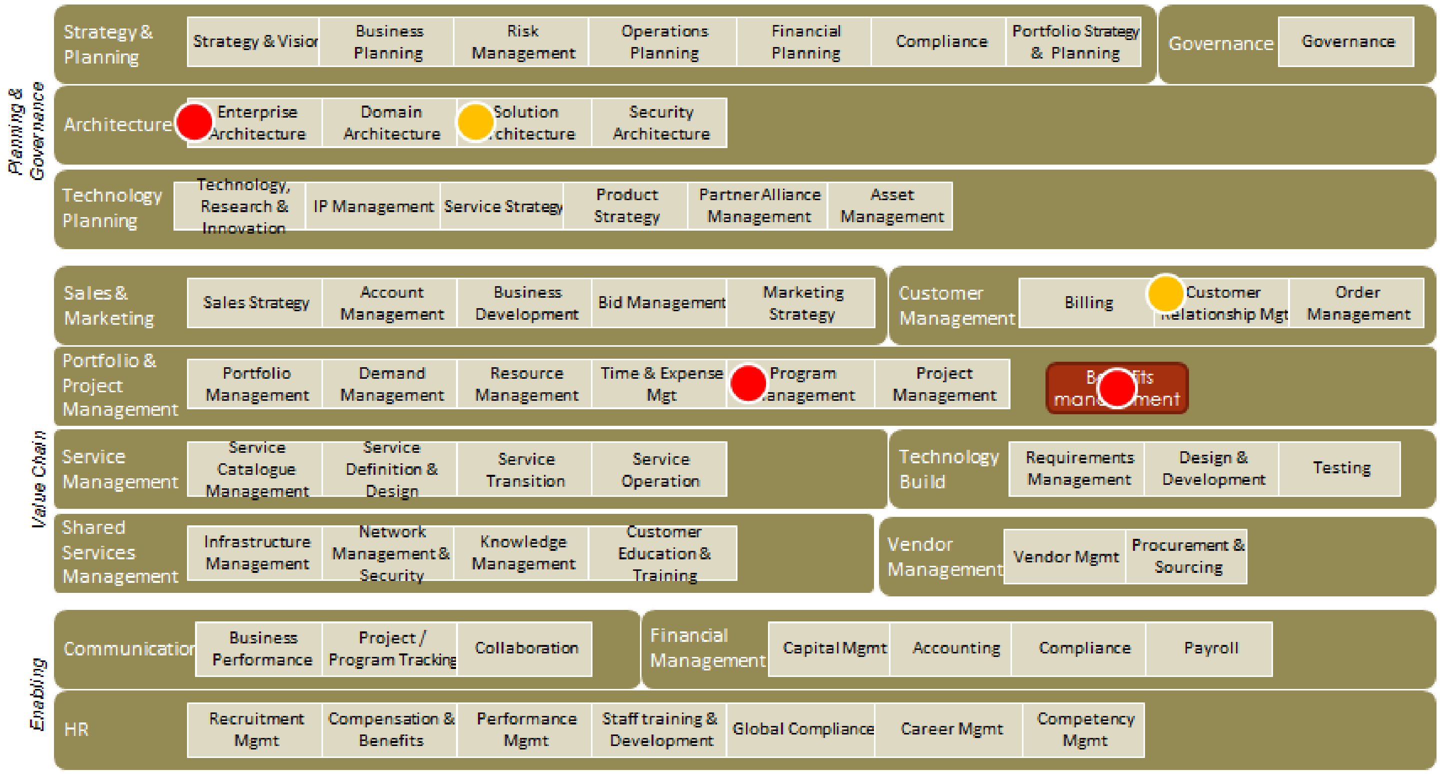


Figure 7

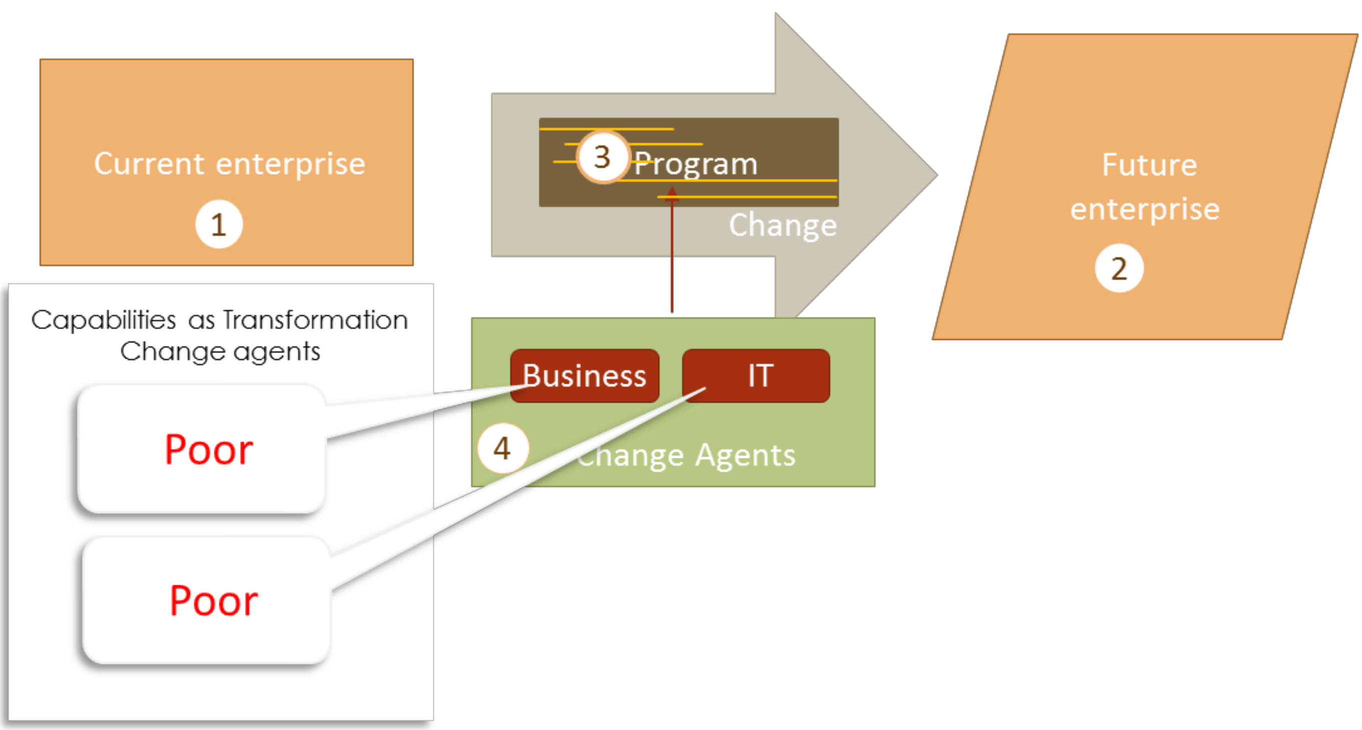


Figure 8

ABOUT THE AUTHORS

Inji Wijegunaratne is an IT consultant with over 30 years' experience, for the past 15 plus years focusing on enterprise architecture. He has held IT architecture management positions in the corporate sector and consulting positions at Deloitte, IBM, and Capgemini. He is currently with Infosys Australia. Inji holds Doctoral and Masters degrees in Information Systems from London University. He has published several papers on enterprise architecture and recently co-authored a book entitled "Enterprise Architecture for Business Success".

Sharma Madiraju advises IT senior executives on strategy, organizational design, enterprise architecture, and IT simplification matters across multiple industry domains. He currently leads the Enterprise Technology Strategy and Architecture practice at Infosys Australia. Sharma received his Master of Engineering Science (Artificial Intelligence) from the University of Melbourne and Master of Business Administration (Strategy) from the Melbourne Business School, Australia.

REFERENCES

McKinsey 2012: Delivering Large-scale IT Projects On Time, On Budget, and On Value, Michael Bloch, Sven Blumberg, Jürgen Laartz, McKinsey Insights, October 2012.

McKinsey 2015a: The Eight Essentials of Innovation, Marc de Jong, Nathan Marston, Erik Roth, McKinsey Quarterly, April 2015.

McKinsey 2015b: How do you Govern in a Disrupted World?, Richard Dobbs, James Manyika, Jonathan Woetzel, McKinsey Global Institute, May 2015.

Sexton 2014: Bungling in State Public Sector Blamed for IT Blowouts, Reid Sexton, The AGE newspaper, Victoria, Australia, November 24, 2014.

Article

Measuring the Quality of Enterprise Architecture Models

Cameron Spence and Vaughan Michell

Abstract

In this article we consider how to measure the quality of a set of Enterprise Architecture (EA) models. We review some relevant literature, focusing in particular on conceptual model quality, and adapt a conceptual model for use specifically with sets of EA models. We develop three objective metrics for this purpose, and also consider the conditions necessary for these metrics to converge towards increasing model quality. We conclude with a partial case study where two of these metrics were used in practice that demonstrates how they can be used.

Keywords

Enterprise Architecture, Modelling, Model Quality, Model Completeness

INTRODUCTION

Context

Enterprise Architecture (EA) is widely used to model and analyze businesses, to a greater or lesser degree depending on geography (Schekkerman 2005; Varnus & Panaich 2009) and its practitioners wield a significant amount of influence (either "final decision-maker" or "great deal of influence") on over $\$10^{12}$ of IT-related spend, according to Gartner (Pettey & Meulen 2013).

EA requires the management of complex data sets to satisfy the needs of its various clients, both in the business and technical domains (Rood 1994). For example, in the business domain we would include information about services provided and the actors involved; in the technical domain we would normally include applications, platforms, and technical infrastructure. This data is assembled and presented in different ways to suit the needs of the stakeholders. For example, some stakeholders will be interested in a financial view; others will be interested in security, or in data replication and duplication. A variety of tools are used to manage the underlying data and build these models, ranging from very rudimentary tools such as basic office productivity software, up to sophisticated software products specifically designed to handle this kind of data, such as those researched by Gartner (Brand2015); for example, Sparx Enterprise Architect (Sparx 2015) and Troux (Troux 2016).

Together, the set of models built and managed by this kind of tool provide a visualization of the data and relationships comprising our "body of knowledge" of the architecture of the enterprise in question.

The Problem

This set of models has to be built up over a period of time by a number of people with differing experiences and perspectives, who may not necessarily share the same ideas about how best to represent certain concepts within models, especially in the early stages of the development where it is not entirely clear what should actually be in the set of models representing the enterprise. The model is a simplification of the actual business and supporting ICT. The models are partly used to make business decisions – for example, between possible solution alternatives – on the basis of best fit to the organization's strategic and tactical goals. Therefore, if the information presented in those models is inaccurate, then the decisions made upon the basis of those models are made on the basis of inaccurate information, and are thus more likely to be sub-optimum. Therefore, in order to avoid making poor decisions based upon inaccurate models, the models need to be accurate.

Measuring and ensuring the overall quality of the set of models is therefore important – but how? For example, someone sponsoring the production of the enterprise models may well be interested in knowing when the initial set of models is complete. How could such a question be answered? This article addresses the question of how we can measure the quality of sets of architectural models.

EXISTING LITERATURE

What is a Model?

Models are generally defined as explicit representations of some portions of reality as perceived by some actor

(Krogstie & Jørgensen 2003). Three "features" of a model can be distinguished, according to Stachowiak, translated from the original German by modelpractice (2012):

Mapping: Models are always models *of something*; i.e., mappings from, representations of natural or artificial originals, that can be models themselves.

Reduction: Models in general capture *not all* attributes of the original represented by them, but rather only those seeming relevant to their model creators and/ or model users.

Pragmatism: Models are not uniquely assigned to their originals *per se*. They fulfill their replacement function (a) for *particular* – cognitive and/or acting, model using *subjects*, (b) within *particular time intervals*, and (c) restricted to *particular mental or actual operations* (Stachowiak 1973).

This suggests a model contains a reduced and hence partial representation of the information available, designed for a particular use at particular times, perhaps to help make particular decisions.

Conceptual Model Quality

Quality attributes of EA are examined in a paper from 2013 (Niemi & Pekkola 2013) that seeks to define the attributes of high-quality EA products and services. The paper defines the quality of EA products and services as the extent to which the products and services meet the needs of the EA stakeholders. See also Bernus (2003) which discusses quality again in terms of outcomes; for example, *efficiency* being defined in terms of conveying the intended meaning; and *completeness* being defined in terms of how the model can be used to create the intended interpretation. These outcomes could perhaps be used to help shape the choice of language used for the modeling products, and the set of viewpoints that sufficiently represent the interests of our particular set of stakeholders, but will not help us in defining the quality of a set of models, *given* a particular choice of modeling language and viewpoints.

We are interested in creating a measure that enables us to calculate, in an objective manner, how close a particular set of EA models is to an ideal state, assuming that the particular choice of modeling language and viewpoints has already been made for us.

Shanks et al. (2003) also suggest criteria for validating conceptual models: semantic accuracy, semantic completeness relative to the focal domain, no semantic conflict in model parts, no redundant semantics. In this article, we are assuming that a set of criteria such as these has already been agreed upon as part of the definition of our "modeling language", and asking how we can measure the quality of a set of conceptual (EA) models as a whole.

Semantic conflict is a major issue for EA analysis as stakeholders and participants frequently use terms in conflicting ways; for example, using different terms to describe what is essentially the same concept (e.g., service and function) as far as those using the terms are concerned.

A framework to measure general conceptual model completeness was developed by Lindland et al. (1994) and subsequently extended by Krogstie et al. (1995). The original paper considered model quality from three dimensions, and the subsequent model extended this to six dimensions taken from the field of semiotics. Although Lindland distinguishes between explicit and implicit statements ($\mathcal{M}_\mathcal{E}$ and $\mathcal{M}_\mathcal{I}$), we will not be making that distinction in this work; we will be dealing purely with explicit statements (models) whose conformance to the language $\mathcal{L}$ can be explicitly tested.

Using these sets, Lindland defines *syntactic quality* as the degree of correspondence between model $\mathcal{M}$ and $\mathcal{L}$. The set of syntactic errors is the set difference:

$$\mathcal{M} \setminus \mathcal{L}$$

This is the set of all statements in the model $\mathcal{M}$ that are not part of the language $\mathcal{L}$. In other words, how much of the model is using the wrong language (syntax).

Semantic quality is the degree of correspondence between model and domain.

If the set difference: $\mathcal{M} \setminus \mathcal{D} \neq \emptyset$ then the model contains invalid statements (i.e., the model makes statements that are incorrect; i.e., not in the domain).

If $\mathcal{D} \setminus \mathcal{M} \neq \emptyset$ then the model is incomplete (i.e., there are elements in our EA domain $\mathcal{D}$ that do not appear in the model $\mathcal{M}$).

Pragmatic quality is the degree of correspondence between model and audience interpretation (i.e., the degree to which the model has been understood).

If $\mathcal{I} \neq \mathcal{M}$ then the model has been misunderstood.

Models in Enterprise Architecture

EA is perhaps an over-specified phrase having many definitions in the literature; however, it deals ultimately with the structure and evolution over time of a business and its supporting technology (ISO/IEC/IEEE 2011).

TOGAF®, an Open Group Standard (The Open Group 2011) explains how models, stored in an architecture repository, use elements from a metamodel to represent the "real-world" enterprise, or perhaps a possible future state of it; and that these models, organized according to stakeholder views, present information designed to help specific stakeholders.

APPLICATION OF CONCEPTUAL MODEL QUALITY TO ENTERPRISE ARCHITECTURE MODELS

Sets of Models

To apply Lindland's theory of conceptual model quality to EA, we are going to define the scope of our conceptual model to be the set of all EA models in our control, stored within an architecture repository, so relating the terms from Lindland's original theory to the terms in our article thus:

Table 1: Adaptation of Terminology for Model Quality

Original Theory	Equivalent in this Article
Statement within the Model	One of the models within the repository
Scope of Conceptual Model	The set of all models within the repository

Using the TOGAF enumeration, we expect our models to conform to some kind of agreed language, and to consist of diagrams, lists, or matrices:

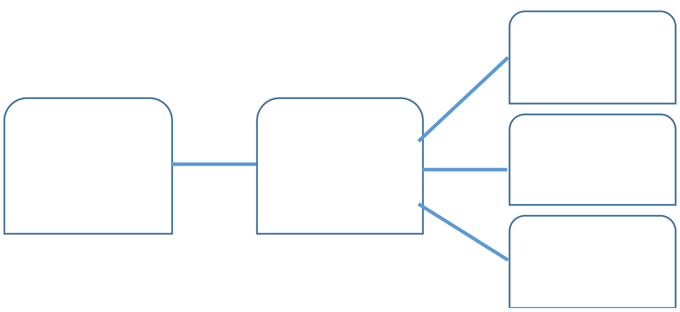


Figure 1: Types of Models

Syntactic errors can be associated with any diagram that does not correspond to our expected language. The language here we will take to be the model types and contents suggested by our architecture framework. Thus, any model in the scope of that framework, that does not use the style of diagrams suggested by that framework, we can define to be syntactically incorrect. That modeling language can be defined as appropriate; it might be a formally defined language such as UML® (Rumbaugh et al. 2004) or the ArchiMate® language (Lankhorst et al. 2009); or we can extend it to cover any type of less formal model that we consider appropriate for our enterprise (perhaps defined in Phase 0 of the TOGAF ADM).

Similarly, we can take each model that we have, and if it does not reflect "reality", either current or planned future, then we define that to be semantically incorrect.

We can therefore adapt Lindland's definitions to this new context:

$\mathcal{M}$ is the set of all the EA *models* within our architecture repository (typically each will be a diagram, catalog, or matrix, as per the TOGAF definition), irrespective of their content (subject) or format; each of the models being a simplified, tailored view of (a possible) reality designed to meet the needs of a particular stakeholder;

$\mathcal{L}$ is the language; i.e., the set of all statements which are possible to make according to the vocabulary and grammar of the EA language(s) that we have agreed to use in our repository; for example, use-case diagrams from UML or structural or behavioral diagrams from the ArchiMate language;

$\mathcal{D}$ is the domain, the boundary of our EA, both the current state of the business and its supporting ICT, and possible future states (options or alternatives) that we are investigating. More precisely, this is the full set of models that we would expect to see in order to fully and accurately describe the domain of interest. In TOGAF terms, this means the set of models that fully populates the required views and viewpoints for all the stakeholders in scope. An example of a model in $\mathcal{D}$ would be a diagram that showed an existing business service (e.g., Intelligence Management, in the policing sector) relating to a new Intelligence system that is being acquired, because this represents a possible (indeed planned) reality;

We are focused on the syntactic (relating to tokens and language) and semantic (relating to meaning) views. The pragmatic elements are out of scope at this stage.

We can visualize the three sets $\mathcal{M}$, $\mathcal{L}$, and $\mathcal{D}$ thus:

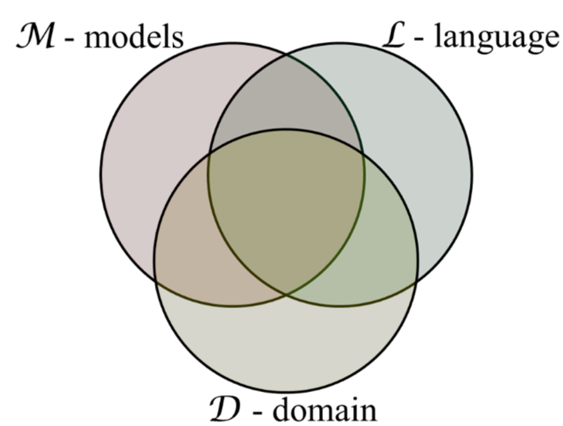


Figure 2: Intersection of Models, Language, and Domain

An assumption we are making at this stage is that anything in the domain $\mathcal{D}$ is in theory capable of being modeled in language $\mathcal{L}$. That is, $\mathcal{L}$ is sufficiently comprehensive to be capable in principle of modeling the whole of $\mathcal{D}$.

Each individual model m within the overall set $\mathcal{M}$ can be considered as a point within one or more of the circles. It is valuable to consider the precise meaning of each of these regions, as they will have a bearing on our quality metrics defined later on.

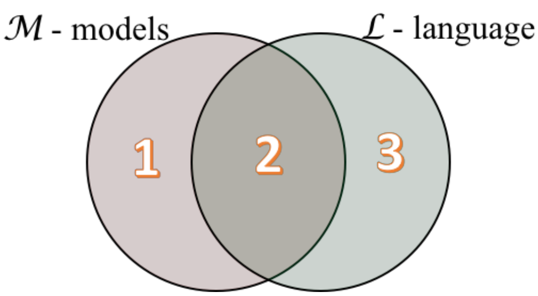


Figure 3: M and L Regions

Region *1* represents existing models (diagrams, matrices, or catalogs) that are not compliant or written within the agreed modeling notation (language) for our enterprise, or for a particular architecture repository.

Region *2* represents existing models that are compliant and written as (syntactically) correct statements in the modeling language.

Region *3* represents models, or types of models, that would be compliant with our language $\mathcal{L}$, but which have not been created (perhaps because they are not required). An example of this might be UML interaction diagrams, if our agreed notation is UML but we have only hitherto used (say) sequence and use-case diagrams. We cannot in this diagram indicate relevance to the real world because the domain $\mathcal{D}$ does not appear in it.

Region *4* represents existing models that do not reflect the domain, either its current state or a possible future state. An example of this might be a model that was created some time ago, reflecting what was deployed previously, but which is now out of date because the environment has since changed. Models in this region (or subregions thereof) we classify as inaccurate, so count against our accuracy metric Q_A, defined later.

The modeling language or syntax irrelevant to regions *4*, *5*, and *6* because the modeling language $\mathcal{L}$ is not included in this diagram.

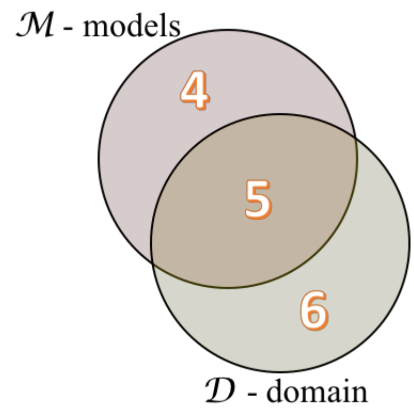


Figure 4: M and D Regions

Region *5* represents existing models that do accurately reflect what currently exists, or a possible option, which may or may not be compliant with our modeling language.

Region *6* represents models that should exist (i.e., would be required to fully populate the required views and viewpoints), but do not. For example, if we have decided that, as a standard, we should have a particular kind of model for every business service, explaining the purpose of the models, its various stakeholders, and the resources that it requires, then every model that exists, we would put in region *5* (assuming that it is accurate), and every business service model that is missing, we would put in region *6*.

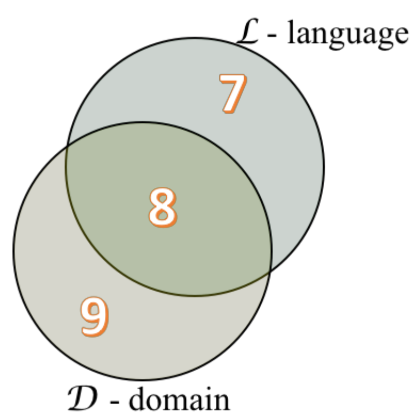


Figure 5: L and D Regions

Region *7* represents existing or possible models (note that the set of models that actually exists, $\mathcal{M}$, is excluded from this diagram) that are compliant with our agreed language $\mathcal{L}$, but which are outside our domain of interest.

Region 8 represents models (existing or possible) within our domain of interest (actual or possible reality) that are also compliant with $\mathcal{L}$.

Region 9 represents models (existing or possible) within our domain of interest that are in theory capable of being represented in a form compliant with $\mathcal{L}$, but which are not in fact compliant with $\mathcal{L}$. This assumes again that all relevant truth about $\mathcal{D}$ can be modeled in $\mathcal{L}$; we consider how to deal with exceptions to this in a later section.

Further meaning can be elaborated for the smaller regions (for example, the intersection of $\mathcal{M}$, $\mathcal{L}$, and $\mathcal{D}$), but they are not necessary for our analysis here.

Models in these regions are summarized mathematically as:

Table 2: Mathematical Definitions of Model Regions

Region	Definition	Region	Definition
1	$m \in M \setminus L$	6	$m \in D \setminus M$
2	$m \in M \cap L$	7	$m \in L \setminus D$
3	$m \in L \setminus M$	8	$m \in L \cap D$
4	$m \in M \setminus D$	9	$m \in D \setminus L$
5	$m \in M \cap D$		

Defining Quality Metrics

We can use this approach to define three quality metrics for sets of architecture models, adapting Lindland's method.

Syntactical Quality

We define the syntactical quality of the overall set of EA models by comparing the total number of correctly formatted models that exist (i.e., models conformant with language $\mathcal{L}$) that exist with the total number of models (irrespective of format or syntax), considering therefore the ratio of the populations of region 2 and $\mathcal{M}$:

$$Q_S = \frac{|M \cap L|}{|M|} \qquad (1)$$

In other words, this is the number of EA models corresponding to our defined language $\mathcal{L}$, divided by the number of models. Obviously:

$$0 \leq Q_S \leq 1$$

If $Q_S = 1$ then all of the models are as defined by our modeling language. We discuss later a possible extension to this definition.

Semantic Quality

Similarly, we can consider a different intersection to determine whether our models are truthful, by defining the semantic quality, or Accuracy, of the overall set of EA models. This would use as the numerator the population of region 5 thus:

$$Q_A = \frac{|M \cap D|}{|M|} \qquad (2)$$

In other words, this is the number of models corresponding to our defined domain $\mathcal{D}$, divided by the number of models. Obviously:

$$0 \leq Q_A \leq 1$$

If $Q_A = 1$ then all of the models accurately reflect our domain; that is, they all reflect current reality, or a possible reality (may make sense in the Opportunities and Solutions phase of the ADM when we consider a variety of possible implementations).

Completeness

This metric is used when considering whether the set of models completely reflects the domain in question. In other words, have we done enough modeling to satisfy all of our stakeholders, with their different viewpoints? Do we have all the models that we would expect (or need) to have? We define the completeness of the overall set of EA models by comparing the population of region thus:

$$Q_C = \frac{|M \cap D|}{|D|} \qquad (3)$$

In other words, this is the number of models corresponding to our defined domain $\mathcal{D}$, divided by the number of models that we would expect to find in our domain. Obviously:

$$0 \leq Q_C \leq 1$$

If $Q_C = 1$ then we have completely modeled our domain.

These three measures – Q_S, Q_A, and Q_C – provide quantitative metrics that enable us to measure the completeness and quality of a set of EA models with a reasonable degree of objectivity.

Of course, an absolutely ideal set of EA models is one where:

$$Q_S = Q_A = Q_C = 1$$

Quality Convergence

Also of interest is the set of conditions that need to be satisfied in order to make positive progress in increasing the quality of our set of models. If we differentiate these definitions with respect to time, using the quotient rule:

$$\frac{d}{dt}\left(\frac{f(t)}{g(t)}\right) = \frac{f'(t).g(t) - f(t).g'(t)}{g(t)^2}$$

then we obtain the following equations showing how these quality measures change over time.

Convergence of Syntactical Quality

Applying the quotient rule to (1) we obtain:

$$\frac{d}{dt}Q_S = \frac{|M \cap L|'.|M| - |M \cap L|.|M|'}{|M|^2} \qquad (4)$$

But in order for this to be positive (i.e., for Q_S to increase over time), knowing that $|M|^2$ will always be positive, we require the numerator also to be positive, thus:

$$|M \cap L|'.|M| - |M \cap L|.|M|' > 0 \qquad (5)$$

Now if no new models are being created, and all effort is directed for a time into modifying the existing models to correct their syntax, then $|M|'$ will be zero, and so from inspection, if any models are being modified to comply with the language L then $|M \cap L|'$ will be positive, and thus Q_S will inevitably increase over time. However, if people continue to create new models with the incorrect syntax, then it will be much harder for the overall syntactical quality to increase. We should then require, rearranging (5):

$$\frac{|M \cap L|'}{|M|'} > \frac{|M \cap L|}{|M|} \qquad (6)$$

Put into words, the ratio of the rate of change of correctly formatted models to the rate of change of all models needs to be greater than the ratio of the number of correctly formatted models to the number of all models. If there are a low percentage of correctly formatted models, then it will be relatively easy to achieve this inequality. For example, if only 1 in every 10 models is correctly formatted, then we only require

$$\frac{|M \cap L|'}{|M|'} > 0.1$$

in order to make progress, which means that the ratio of new correctly formatted models to the number of new incorrectly formatted models just needs to be more than this (relatively low) target. We can see, however, that as our overall model quality increases, it would be relatively easy for the overall syntactic quality to slip backwards as we require the above ratio to be higher and higher, approaching 1.

Convergence of Semantic Quality

Applying the quotient rule to (2) we obtain:

$$\frac{d}{dt}Q_A = \frac{|M \cap D|'.|M| - |M \cap D|.|M|'}{|M|^2} \qquad (7)$$

But in order for this to be positive (i.e., for Q_S to increase over time), knowing that $|M|^2$ will always be positive, we require:

$$|M \cap D|'.|M| - |M \cap D|.|M|' > 0 \qquad (8)$$

Now if no new models are being created, and all effort is directed for a time into modifying the existing models to make them accurate (i.e., describe what actually exists, or might (a solution option)), then as with syntactical quality, $|M|'$ will be zero, and so from inspection, if any models are being modified to make them accurate, then $|M \cap D|'$ will be positive, and thus Q_A will inevitably increase over time. However, if people continue to create new models that do not reflect [a possible] reality, then it will be much harder for the overall semantic quality to increase. We should then require, rearranging (7):

$$\frac{|M \cap D|'}{|M|'} > \frac{|M \cap D|}{|M|} \qquad (9)$$

Put into words, the ratio of the rate of change of accurate models to the rate of change of all models needs to be greater than the ratio of the number of accurate models to the number of all models.

As discussed in the previous section on syntactical quality, this task may get harder as the overall semantic quality of the model [set] increases.

Convergence of Completeness

Applying the quotient rule to (3) we obtain:

$$\frac{d}{dt}Q_C = \frac{|M \cap D|'.|M| - |M \cap D|.|M|'}{|D|^2} \qquad (10)$$

But in order for this to be positive (i.e., Q_S should increase over time), knowing that $|D|^2$ will always be positive, we require:

$$|M \cap D|'.|M| - |M \cap D|.|M|' > 0 \qquad (11)$$

This is of course exactly the same condition that we require for the semantic quality to increase; the choice of denominator (number of models that exist, $|\mathcal{M}|$, or number of models that should exist, $|\mathcal{D}|$) is not determinative of the direction of travel because it is squared and thus always positive. The denominator just affects the magnitude of the rate of change of the quality measures. The more models that *actually* exist, the slower it will be for a given modeling effort, to increase Q_A, whereas the more models that *should* exist, the slower it will be to change Q_C.

EXTENSION FOR EVOLVING MATURITY

For organizations that are relatively new to architecture frameworks, one further extension of our calculations may be helpful. There may be occasions when modeling efforts start without having a clear scope or framework in mind. Later on, when the scope has become clearer, our understanding of the appropriate language for our models may become clearer.

However, in the early stages of modeling, our language may not be complete. In other words, there may be statements of truth in domain $\mathcal{D}$ which are not actually expressible in language $\mathcal{L}$, which corresponds to regions in Figure 2 above, which are inside $\mathcal{D}$ but outside $\mathcal{L}$.

This might apply in particular to free-format models that are designed in particular for business users for whom formal languages (such as UML) would be inappropriate. Another example might be a desire to incorporate process models with swimlanes in our models, where we have restricted ourselves at the current time to using only the ArchiMate language which does not support these kind of diagrams.

Now we could just extend our language definition $\mathcal{L}$ to include such free-format models. However, we feel that it is useful to recognize this potential disconnect between what we want to be able to model (domain $\mathcal{D}$) and our understanding of the best way of modeling this (modeling language $\mathcal{L}$). Although the original concept as discussed in Lindland did not allow for this, we feel this is a useful extension for situations where the modeling language is still evolving to take into account more and more of an existing EA model content. The EAF^2 (Franke et al. 2009) may be useful when comparing EA frameworks to see which of them are able to model what kind of concepts.

Thus, when we count the number of models in $\mathcal{M}$, when calculating the syntactical quality of the set of models, we may wish to exclude those that are not currently expressible (whose format is not prescribed) in $\mathcal{L}$. This will have a direct effect on how we actually assess the quality of an EA model.

However, with our current definition of Region 9, there is no way of making this distinction (between models that should exist, but are outside the bounds of our language, and those that should exist, and are not). Recall that our definition of this region is that it contains all models that should exist (in our domain $\mathcal{D}$) that are expressible in language $\mathcal{L}$, but are not in fact conformant to $\mathcal{L}$. We can in principle consider a subset of all models $\mathcal{M}$ that are currently outside the scope of our formal language $\mathcal{L}$. That is, they cannot be expressed in $\mathcal{L}$ even if we wanted to, and even though they describe part of our domain $\mathcal{D}$.

We can say that:

- Models falling inside the scope of language $\mathcal{L}$ comprise set $\mathcal{M}_{S}$.
- Models falling outside the scope of language $\mathcal{L}$ comprise set $\mathcal{M}_{O}$.
- The whole set of models, $\mathcal{M} = \mathcal{M}_{S} \cup \mathcal{M}_{O}$ (the union of the two sets).
- A model is either inside or outside the language, so $\mathcal{M}_{S} \cap \mathcal{M}_{O} = \emptyset$ (none are in both).

Thus, for syntactical quality, we are only concerned about models that *should* conform to the language but *don't*, so we need to revise definition (1) above to give:

$$Q_S = \frac{|M \cap L|}{|M_S|} \quad (12)$$

The time derivative of this is given by:

$$\frac{d}{dt} Q_S = \frac{|M \cap L|'.|M_S| - |M \cap L|.|M_S|'}{|M_S|^2} \quad (13)$$

and in a similar manner to before, we can see that for this to converge towards 1, we should require:

$$\frac{|M \cap L|'}{|M_S|'} > \frac{|M \cap L|}{|M_S|}$$

Summary of Quality Metrics

We have constructed three normalized measures of the quality and completeness of an overall EA model, related to syntax, truthfulness (accuracy), and completeness, and also looked at how they change over time:

Table 3: Summary of Quality Metrics

Metric	Rate of Change of Metric
$Q_S = \frac{|M \cap L|}{|M_S|}$ (12)	$Q_S' = \frac{|M \cap L|'.|M_S| - |M \cap L|.|M_S|'}{|M_S|^2}$ (13)
$Q_A = \frac{|M \cap D|}{|M|}$ (2)	$Q_A' = \frac{|M \cap D|'.|M| - |M \cap D|.|M|'}{|M|^2}$
$Q_C = \frac{|M \cap D|}{|D|}$ (3)	$Q_C' = \frac{|M \cap D|'.|M| - |M \cap D|.|M|'}{|D|^2}$ (10)

PARTIAL CASE STUDY

Context for Case Study

The study focused on a UK public sector organization where development teams were engaged in a number of delivery projects, documenting their design work in a single enterprise-context syntactical Wiki.

The lead author had been asked to recommend a new structure to be used by these teams, in effect a new architecture framework and a corresponding standard layout for Wiki pages, one page per architectural entity. The question was asked: "how close are we to having the pages complete and in the new layout?".

This provided the motivation to construct an objective methodology for measuring the completeness of a set of models which could then be used in practice in this particular organization to answer the question posed. A "model" in this context we chose to relate to a complete Wiki page. The intention was that each Wiki page should correspond to a single architectural entity (e.g., a particular business service or solution building block). Thus, our definition of the approved language $\mathcal{L}$ in this scenario included all standard elements that we would expect to see on a standard page for (say) a business service, including a diagram giving it context as well as standard header and footer information giving related information (for example, any information objects used by it – so relationships to other entities).

Counting the Number of Models

The number of models is one of the key figures used to calculate our metrics derived above. Recall that for the syntactical measure Q_S, we are interested only in models that are within the scope of our modeling language $\mathcal{L}$ (our new standard layout, using new architectural terms), whereas for the other quality measures we are interested in all models. When we carried out this exercise, the total number of pages (available from a database report) in our Wiki was 6,598; this is the value we used for $|\mathcal{M}|$.

For Qs, though, we counted how many pages held content that should comply with $\mathcal{L}$, which we will interpret to mean how many pages hold content in the scope of our new framework ($\mathcal{M}_s$, not $\mathcal{M}$). Given that we were changing the architectural framework, and in particular introducing some new entities not previously used in this organization, we interpreted this to mean "how many pages hold information describing entities that appear in the new entity metamodel?". Taking a sum of the pages that held such information, we arrived at a total of 2,738 for:

$$|\mathcal{M}_s| \qquad (14)$$

Counting Pages with the Correct Syntax

Only 328 pages were in the correct format. This is a low percentage of the total number of pages, but was to be expected as we were in the very early stages of rolling out a change to the way that we described our EA.

Thus, we used a value of 328 for:

$$|M \cap L| \qquad (15)$$

Counting Completeness of Models

We were able to do this, although the figures involved are very approximate, due to the uncertainty over the expected population of region $\mathcal{D}$ which requires us to consider all models that we would expect to find. In our situation, we did this by means of the following algorithm:

```
SET expected page count to 0
FOR each entity in the new metamodel
    IF a complete set of instances
    exists for that entity
    THEN
        Add the number of instances
        to our expected page count
    OR
        Approximate the number of
        instances by reference to
        another entity
        Add that approximation to our
        expected page count
    ENDIF
END LOOP
```

We ran reports on the Wiki using the above algorithm manually to calculate an approximation to our expected page count. By way of example, leaving aside the contextual layer which we used for overarching governance information such as principles, we assigned entities in our metamodel within one of three layers: conceptual, logical, and physical; and for Information Architecture in particular, we had one entity in each: a conceptual information entity (which we called a "Business Object"), a logical information entity ("Logical Entity"), and a physical information entity ("Physical Entity").

When it came to counting how many models we expected in the information domain, we already had a complete set of Business Objects, and so we just used the total number of Business Objects to add to our total of expected pages (one page per entity instance). However, for the next layer down, Logical Entities, we did not have a complete set. Therefore, we used the examples where we had populated this layer to estimate how many Logical Entities there were likely to be, on average, for each Business Object. This enabled us to estimate a likely total number of Logical Entity pages that we expected. Obviously, the more complete sets of entities exist, the more accurate this kind of estimation would be.

Across the complete set of entities, we found that whereas we would have expected to see 1,524 models (one for each entity instance, if fully populated), in fact we could only see 674:

$$|\mathcal{D}| = 1524 \quad (16)$$

$$|M \cap D| = 674 \quad (17)$$

Assessment of Current Quality

We were then in a position to calculate two of our three quality measures for the current set of architecture models.

Syntactical Quality

Using equation (12), and inserting values from (14) and (15) for the numerator and denominator respectively, we obtain:

$$Q_S = \frac{328}{2738} = 0.12, \text{ or } 12\%.$$

So we can say that, in terms of syntax, our overall set of models are 12% of the way to being correct. It should not be difficult to increase this quality measure, because the condition that needs to be satisfied for the time derivative to be positive, from (6), is:

$$\frac{|M \cap L|'}{|M|'} > 0.12$$

In other words, if the number of correct models being produced per unit time is more than 12% of the total number of models being produced (that is, if more than 1 out of every 8 new models is correctly formatted), then we will make progress in increasing the overall model quality with regard to its syntax. We will obviously make progress much faster if all new models created are of the correct syntax – that is, if our development teams stop creating models that use the old syntax.

Completeness

Using equation (3), and inserting values from (17) and (16) for the numerator and denominator, we obtain:

$$Q_C = \frac{674}{1524} = 0.44, \text{ or } 44\%.$$

So, in terms of completeness, we can describe our overall set of models as being 44% complete.

We will need to work a bit harder now to maintain this completeness figure, because from (11), we can see that we require:

$$\frac{|M \cap D|'}{|M|'} > 0.44$$

So, if we are expanding the set of models that should exist, then we need to ensure that we actually produce over 44% of what is being asked of us in order for us not to fall behind. As with all these quality figures, of course, we will make progress much faster if we stop producing models with incorrect syntax (to help Q_S) and make concerted efforts to finish the required modeling before increasing the need for more models (to help Q_C).

CONCLUSIONS

We have demonstrated that it is possible to create objective metrics to enable us to measure different aspects of the quality of a set of EA models. We can measure the quality of their syntax; the quality of their meaning; and their completeness.

REFLECTIONS

We were unable to test the accuracy metric Q_A for practical reasons, although in theory there is no reason why this could not be done given more time. It will be harder to test that "model represents reality" where either:

- Reality is abstract and/or subjective (e.g., "what is our set of business services" is a more subjective question than "what is the structure of that database"), or
- Reality is a possible reality – one of the options being explored as part of the Solutions and Opportunities phase.

Two other concepts from the Lindland paper have not been incorporated so far. Firstly, that paper has the concept of "feasible completeness", recognizing that it may not be worth the effort completing 100% of the model. The incorporation of this concept would affect our target of 1.0 for one or more of our metrics.

Secondly, the Lindland paper considers the pragmatic level – how content from the model is interpreted by those perceiving it. This is surely relevant to EA models which are intended to communicate to EA stakeholders.

There are perhaps ways that these difficulties could be addressed. External industry-specific standards can provide reference sets and catalogs of some business and technical artifacts (the TOGAF Technical Reference Model is an example of one of these), against which our models can be compared.

The set representation seems to be self-consistent as developed so far, although it would be preferable perhaps to be able to incorporate the concept of domain knowledge outside the scope of the language in our sets somewhere (the distinction between $\mathcal{M}$ and $\mathcal{M}_l$).

FURTHER WORK

Further research would be useful to explore a number of ideas, some of which are discussed above:

- How can we be more certain about the accuracy of models corresponding to either abstract concepts, or realities that do not yet exist?
- How do we better incorporate domain knowledge outside the scope of our modeling language into our formal description?
- How can we incorporate the idea of "feasible completeness" in some objective manner, so we can determine when it is not worth pursuing any further increase in our quality metrics?
- Can we make some kind of relationship between the quality of our EA models and the corresponding quality of decisions made upon the basis of those EA models?

This latter question seems particularly relevant, because if we knew that better models led to better decisions, then that gives us a stronger motivation to produce better models.

ABOUT THE AUTHORS

Cameron Spence works as an Enterprise Architect for Capgemini in the UK, specializing in information and application architecture, portfolio management, and IT strategy. He has previously worked for major consulting organizations including Digital, Compaq, Hewlett-Packard, Capita, and Wipro, and is currently working towards a PhD in Enterprise Modeling.

Vaughan Michell is Business Technology Consulting Programme Director within the Informatics Research Centre at Henley Business School. He has a BSc in mechanical engineering from UCL, an MBA from Warwick University, and a Robotics D.Phil from Oxford University. Research interests include enterprise architecture, human and machine capability, and knowledge-intensive processes.

REFERENCES

P. Bernus: Enterprise Models for Enterprise Architecture and ISO9000:2000, Annual Reviews in Control (27:2), pp.211-220 (2003).

S. Brand: Magic Quadrant for Enterprise Architecture Tools, Gartner (2015).

U. Franke, D. Hook, J. Konig, R. Lagerstrom, P. Narman, J. Ullberg, P. Gustafsson, M. Ekstedt: Eaf2 – A Framework for Categorizing Enterprise Architecture Frameworks, Software Engineering, Artificial Intelligences, Networking, and Parallel/Distributed Computing (SNPD '09), 10th ACIS International Conference, pp.327-332 (2009).

ISO/IEC/IEEE 42010:2011: Systems and Software Engineering — Architecture Description.

J. Krogstie, H. Jørgensen: Quality of Interactive Models, in Advanced Conceptual Modeling Techniques, A. Olivé, M. Yoshikawa, E.K. Yu (Eds.), Springer Berlin Heidelberg, pp.351-363 (2003).

J. Krogstie, O.I. Lindland, G. Sindre: Defining Quality Aspects for Conceptual Models, ISCO, pp.216-231 (1995).

M.M. Lankhorst, H.A. Proper, H. Jonkers: The Architecture of the ArchiMate® Language, in Enterprise, Business-Process and Information Systems Modeling, Springer, pp.367-380 (2009).

O.I. Lindland, G. Sindre, A. Solvberg: Understanding Quality in Conceptual Modeling, Software, IEEE (11:2), pp.42-49 (1994).

Modelpractice: General Model Theory by Stachowiak (2012); refer to: https://modelpractice.wordpress.com/2012/07/04/model-stachowiak/ (retrieved December 7, 2015).

E. Niemi, S. Pekkola: Enterprise Architecture Quality Attributes: A Case Study, System Sciences (HICSS), 46th Hawaii International Conference: IEEE, pp.3878-3887 (2013).

C. Pettey, R.v.d. Meulen: Gartner Says Enterprise Architecture Practitioners Significantly Influenced $1.1 Trillion, Stamford, Connecticut: Gartner (2013).

M.A. Rood: Enterprise Architecture: Definition, Content, and Utility, Third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp.106-111 (1994).

J. Rumbaugh, I. Jacobson, G. Booch: Unified Modeling Language Reference Manual, Pearson Higher Education (2004).

J. Schekkerman: Trends in Enterprise Architecture 2005: How are Organizations Progressing?, Institute for Enterprise Architecture Developments, Amersfoort (2005).

G. Shanks, E. Tansley, R. Weber: Using Ontology to Validate Conceptual Models, Communications of the ACM (46:10), pp.85-89 (2003).

Sparx Systems: EA Pricing & Purchasing (2015); refer to: www.sparxsystems.eu/enterprisearchitect/ea-pricing-purchasing/ (retrieved March 27, 2015).

H. Stachowiak: Allgemeine Modelltheorie, Wien: Springer-Verlag (1973).

The Open Group: TOGAF®, an Open Group Standard, Version 9.1 (2011).

"Troux Architect (2016); refer to: www.troux.com/enterprise-architecture/products-architect (retrieved March 28, 2015).

J. Varnus, N. Panaich: TOGAF® 9 Survey Results Presentation, The Open Group (2009).

Association of
Enterprise Architects®