



ENTERPRISE ARCHITECTURE PROFESSIONAL JOURNAL

JULY 2017

IN THIS ISSUE

Editor's Welcome

by Darryl Carr, EAPJ Editor

Page 2

Founder's Note

by Dr. Steve Else, EAPJ Founder

Page 3

Enterprise Architecture and Data Modelling

by erwin inc.

Page 4

What Can We Do with the Archimate® Language

by Iver Band, Chair – Archimate Forum at The Open Group

Page 9

Case Study: Archimate® Modelling at Maersk

by Carl Chilley, Head of Strategic Architecture – A.P. Moller Transport & Logistics
and Marc Lankhorst, Managing Consultant – BiZZdesign

Enterprise Architecture Events

Page 31

Call for Submissions

Page 33

EDITOR'S WELCOME

by Darryl Carr, EAPJ Editor

Welcome to the July 2017 Edition of the Enterprise Architecture Professional Journal. We serve practicing and aspiring enterprise architects, as well as those who apply the holistic perspective of enterprise architecture to other disciplines. EAPJ informs their daily work and benefits their careers with content that is focused, concise, authoritative, practical and accessible.

This issue focuses on modelling, tools and the ArchiMate® language.

We open this edition with a look at the importance of supplying quality data to your organization's management team. This material comes from the good folk at erwin, and highlights that data is the lifeblood of the modern enterprise. With an expected dramatic increase in the volume of data available to organizations through current and planned initiatives such as Big Data, Process Automation, the use of Artificial Intelligence and the Internet of Things (IoT), the ability to source and interpret data quickly and effectively is something that organizations need to embrace and embed in all aspects of their operation. An enterprise architecture function that brings together this information to enable better decision making is primed for making a positive impact and realizing value.

Following that, we are very fortunate to have an excellent piece from Iver Band, the recently appointed Chair of The Open Group's ArchiMate® Forum. Iver provides us with a fantastic overview of version 3.0 of the language, using a realistic case study to allow us to explore the power of the language, and some of the great new features built into the latest version. With Iver's insight, we see how to ensure that work being done to introduce new capabilities, processes, systems and technology can be firmly attributed back to a company's goals and strategies, or, as in Iver's case study, how you can balance the interests of two organizations working toward a common goal.

To finalize this issue, and to reinforce the points made in the earlier articles, we are extremely pleased to feature a real-life case study from A.P. Moller – Maersk, detailing how they have used architecture, modelling tools, and the ArchiMate® language to help them with a major piece of transformation work. This article is co-authored by A. P. Moller's Head of Strategic Architecture, Carl Chilley, and BiZZdesign's Managing Consultant, Marc Lankhorst (part of the original team that developed ArchiMate® and author of the Enterprise Architecture at Work books). It highlights how Carl's team have established an impressive balance between architectural rigour, and the pragmatic need to deliver value to the organization in a timely manner.

Back again in this edition of the Journal is also the EA Events section, highlighting some of the activity happening in EA-related events and conferences going on around the world. There is plenty of activity going on, so have a look, and get out there and get involved.

Please contact me at editor@eapj.org with your questions, comments, ideas and submissions. As always, I look forward to hearing from you!

Darryl Carr

Editor, Enterprise Architecture Professional Journal

Opinions noted herein are those of the authors and do not necessarily represent those of the editors or any other interests. Some articles may be published without attribution, but only if the editors ensure their sources are reliable and knowledgeable. Potential contributors are strongly encouraged to submit material to editor@eapj.org.

© 2017 Enterprise Architecture Professional Journal

ArchiMate® is a Registered Trademark of The Open Group – See more information [here](#).

FOUNDER'S NOTE

by Dr. Steve Else, EAPJ Founder

With the focus of this edition of the Enterprise Architecture Professional Journal being on modelling, tools and the ArchiMate language, I thought I would highlight some of my own thinking and experiences on the topic.

ArchiMate 3.0 is a terrific tool for accelerating the knowledge, skills, and maturity (KSM) of individuals wanting to progress rapidly in the discipline of Enterprise Architecture. Of course, it can be, and is being, used by several large companies. One such company is Arizona Public Service Electric Company (<http://aps.com>), which made it a standard at the start of 2017. As a result, its Enterprise and Solution Architects moved from Visio or PowerPoint to ArchiMate in the stretch of only a few months.

Based on my own discussions with them, the architects at APS are in fact amazing one another with their new ArchiMate views of electricity generation and transmission. In doing so, they are leveraging the full richness of ArchiMate 3.0, which includes a physical layer, permitting the easy modeling of energy and supply chains, as well as full business and IT scenarios.

My own company, EA Principals, includes a primer on ArchiMate 3.0 in all of its TOGAF 9.1 Enterprise Architecture (EA) Certification Classes, as well as in its classes on the Federal Enterprise Architecture Framework. By including ArchiMate 3.0, even at a high level, students can quickly see how seamless the move can be to modeling EA capability and scoping EA initiatives (versus just talking about these aspects in a conceptual and textual way). In fact, we have started using the key elements of ArchiMate 3.0 for Strategy, Motivation, and Implementation & Migration in Mind Maps to help students with a repeatable approach to setting up EA initiatives (prior to even having them learn the icons).

We have a great line-up of articles in this edition of the Journal. We hope you enjoy them.

Dr. Steve Else

Founder, Enterprise Architecture Professional Journal.

ENTERPRISE ARCHITECTURE AND DATA MODELLING

By erwin inc.

Practical steps to collect, connect and share your enterprise data for better business outcomes.

From the Editor:

This article comes to us from [erwin](#), who last year acquired Corso, one of the leading Enterprise Architecture tools in the market. In this article, we hear how leveraging data effectively, supporting better decision making, and taking a business outcome driven approach to Enterprise Architecture, can dramatically improve the traction of the EA function. This, in turn, allows the Enterprise Architect to have a greater influence on the success of the organization.

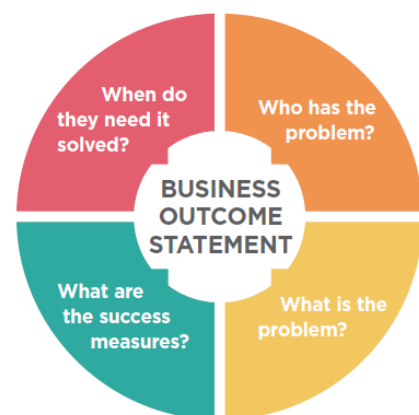
With an expected dramatic increase in the volume of data available to organizations through current and planned initiatives such as Big Data, Process Automation, the use of Artificial Intelligence and the Internet of Things (IoT), the ability to source and interpret data quickly and effectively is something that organizations need to embrace and embed in all aspects of their operation. An enterprise architecture function that brings together this information to enable better decision making is primed for making a positive impact and realizing value.

Thanks go to [Mariann McDonagh](#) and [Martin Owen](#) from erwin Inc. for sharing this material with us.

Introduction

All too often, enterprise architecture has been executed by IT groups for IT groups, and has involved the idea that everything in the current state has to be drawn and modeled before you can start to derive value. This approach has caused wasted effort, taken too long to show results, and provided insufficient added value to the organization. In short, for many organizations, this has led to erosion in the perceived value of enterprise architecture.

Taking a Business Outcome driven approach has led to renewed interest in the value Enterprise Architecture can bring. But such interest will only remain if EA teams remember that **effective architecture is about enabling smarter decisions**, enabling management to make those decisions more quickly, by having access to the right information, in the right format, at the right time. Of course, focusing on future state first (desired business outcome), helps to reduce the scope of current state analysis and speed up the delivery of value. This increases perceived value, while reducing organizational resistance to architecture.



The four key inputs to a well crafted business outcome statement

Use a Data-Driven Enterprise Architecture to Deliver Better Business Outcomes

Taking a business outcome-centric approach means you should:

- a) **Understand who the real stakeholders involved in and benefiting from the enterprise architecture are in your organization.** While many stakeholders may still be within the IT organization, it is the business and C-level stakeholders who should be able to gain the most.
- b) **Understand their goals, objectives and pain points,** and then help them to express them in clear business outcome related terms. This will take time and skill, as many business users simply asking for system changes without clearly stating their actual objectives.
- c) **Review your current architecture efforts and tooling.** Question whether you are providing or managing data the business does not need, whether you are working too deeply in areas that may not be adding value, or whether you have your vital architecture data spread across too many disconnected tools.

Often when architects (or indeed vendors) talk about the C-Suite, they simply refer to the role of CxO, which is a reminder that executives have different needs and agendas to most people in the IT domain. But to think of them as a homogenous group is a mistake.

Only when the connection between architecture and priority is explicitly stated in ways and terms the address the specific concern of the specific “C” role will this become apparent.

| CEO | CFO | CIO | CDO |
|---|-------------------------------|---------------------------------------|---|
| Stronger Client Focus | Cost optimization | Improving business processes | Leverage existing data to advance business |
| Fostering innovation | Process optimization | Reducing costs and driving efficiency | Find new revenue streams by monetizing data |
| Minimizing cybersecurity risks | Cash and liquidity management | Maintaining IT systems | Enrich data by linking with other data |
| Implementing disruptive technology | Financial risk management | Managing cybersecurity | Maintain quality and relevance of data |
| Responding effectively to regulatory change | ERP implementation | Business innovation | Protecting data as an asset |

While the needs of C-level executives may be different, they are all trying to make smarter decisions that enable achievement of their desired outcomes.

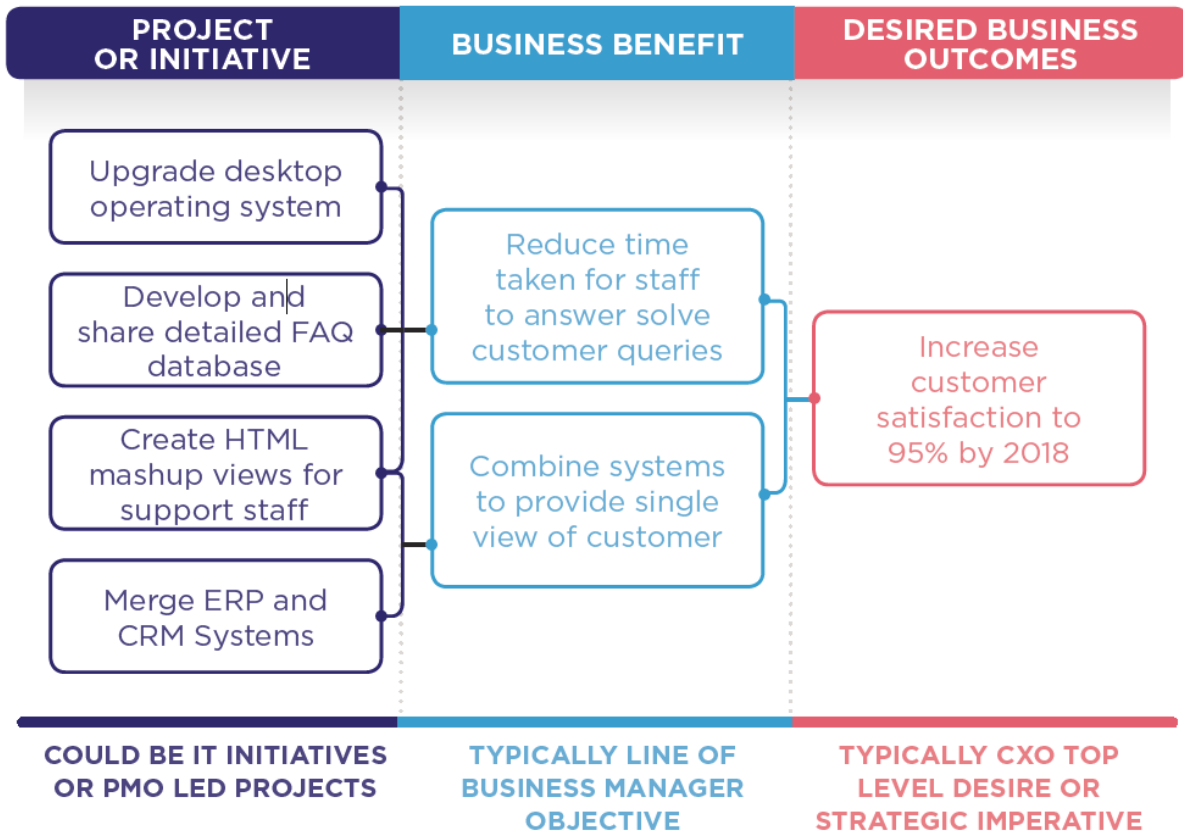
In order to make those decisions, they require actionable insights. As an architecture team, it is up to us to help deliver that insight. When we do, and when we help others connect their projects, programs and initiatives to those outcomes, then everyone understands **why** we need to do the things we do – putting the data in context and thus providing actionable information.

In order to make your architecture efforts relevant and value added from a C-Suite perspective and increase your chances of getting or keeping budget:

- a) **Use benefit trees** to illustrate how your projects or architecture initiatives connect to the desired C-Suite outcome.
- b) **Focus architecture efforts** on presenting the data management needs to make decisions, rather than the maps and models technical staff use. This may mean maintaining multiple

views of the same information, such as a business presentation version and a more technical model-based view.

c) **When collecting or connecting data, think broadly.** A catalogue of processes, a list of standards, or a common dictionary of business terms are all forms of data. Data about the organization is just as important as business performance, customer, product or accounting data that forms part of the everyday fabric of the organization.



An example benefits tree illustrating how a “desktop upgrade” is not an IT project that costs, but a mechanism to help the organization achieve one of its key business outcomes.

Enterprise and Data Architecture teams that fail to fully realize the impact of Digital Business on the IT landscape are putting their organizations at risk

There can be no doubt that the digital business wave will affect businesses large and small, across all industry sectors and all geographies. The disruptions it has caused so far are nothing compared to the disruptions that are coming. While self-driving cars, digital assistants, and drone delivery are headline grabbing, the real impact goes far beyond these new technologies.

The internet of things (IOT) will drive vast amounts of sensor data on a minute by minute or second by second basis. That information will need to be sorted, catalogued and acted upon, often in the blink of an eye. Your applications and systems may no longer serve people but instead respond to requests from other systems. These systems will not be able to correct mistakes or be willing to wait a few more minutes while you work on things. The inability to respond accurately in real time may well spell the death of your business, and responding to the challenge will require close coordination between data management and enterprise architecture teams.

As organizations seek to leverage data to create and sell new digital products, often via APIs, enterprise and data architects will need to learn the skills of product managers. Once data or an API becomes a commercial product, new rules apply when it comes to quality, regularity of updates, timeliness of availability. You will be required to take a customer-first approach, and understand

revenue risks. You may also need multi-lingual documentation, product support functions, and ways to deliver these new digital products. In these respects, neither data management or enterprise architecture can afford to work alone – the risk of historical myopic thinking is just too great.

Closer to home, enterprise architects and data modelers are wrestling with the fact that much of their hitherto closed data may soon need to be exposed as open data via APIs. Where in the past systems and the data processed have all been within the control of the IT groups, it is now increasingly held in third party applications based in the cloud or via APIs from other members of your ecosystem.

The interactions and integrations between applications, inside and outside the firewall, means that enterprise architecture and data modeling teams must work closer together. They need to coordinate more effectively to ensure that data quality and data security is given greater emphasis, that they can collect and coordinate data that may not all be under their direct control, and be capable operating not just as managers in IT but as product managers too.

Enterprise architecture and data management teams should:

- a) **Get together to discuss** what digital initiatives are already underway or planned, and assess how they might be held back by current data practices.
- b) **Work to ensure** that, in addition to solid physical model documentation for maintenance, there are comprehensive logical and conceptual models available for all parties affected by digital business. This may include business analysts, systems analysts, solution architects, business architects and, if your organization has one, the Chief Digital Officer.
- c) **Be proactive** in identifying ways that their organizations might package data to create new digital products, either on their own as part of another ecosystem.

Enterprise Architects that focus more on the outputs and audience for architecture increase their standing and value to the organization



Architecture is data too, and needs to be exercised to justify keeping it up to date and relevant. While the creation of architecture might reside with specialist teams, the value only comes when broad audiences within the organization use it and come to rely on it. While we have been talking about addressing the needs of the C-Suite and delivering the information they need in a timely manner, there is another group we need to consider. That group includes everyone in the IT organization, from architects to designers, from relationship managers to business analysts, from ERP teams to infrastructure teams. All these people are undertaking work, making decisions and implementing

activities that are informed by and impact your enterprise architecture. To be credible with the non-IT audience, we first should demonstrate common use of the architecture within our organization.

Just as with non-IT users, this broad acceptance of architecture will only happen if the architectural information that is shared is relevant, easy to access and current. Talk to the different constituencies

who could or should use the information, and ensure the delivery of architecture information that is tailored to suit their needs.

- a) **Socialise your architecture efforts, such that they become the business decision tool of choice for the entire organization.** This applies not just to disseminating information, but in collecting information too. The more people are heard, then the more they will listen.
- b) **Take time to understand the information needs of different roles within the IT organization,** and present that information to them such that the enterprise architecture becomes a powerful tool to effectively do their job.
- c) **Do not force architectural views on people who do not need it.** Instead, accept that multiple views on data for different purposes is not only OK, but to be welcomed if it increases use of architectural data.

A Last Thought

This article has focused on practical steps and actions that will enable you to collect, connect and share the right data at the right time to increase your chances of success. The steps outlined here will also enable you gain greater acceptance of architecture and its value within your organization, but there is one other important point for you to consider.

Much of the work in modern IT groups focuses on application portfolio management, de-duping overlapping systems and reducing the number of suppliers. To be credible, IT groups need to do more to apply the same principals themselves. For example, you can't use 5 different vendors and tools for modeling and architecture using the argument that every individual group is different! You can't talk credibly to the business about consolidating data, while holding IT data in more systems than you need. And you can't justify to the business why they need to change their processes, or switch off their favorite systems, and try new ways of working – while you are not willing to change yourself. So consider how you can consolidate your modeling and architecture tools, the way you approach architecture, and how to achieve pragmatic, data driven, outcome based approaches that elevate the perceived value of your architecture initiatives.

WHAT CAN WE DO WITH THE ARCHIMATE LANGUAGE

by Iver Band, Chair – Archimate Forum at The Open Group

June 24, 2017

Background: What is the ArchiMate Language?

Last year, the Open Group released version 3.0 of the ArchiMate® standard [1], which provides a language with concepts for describing enterprise and solution architectures, a framework (

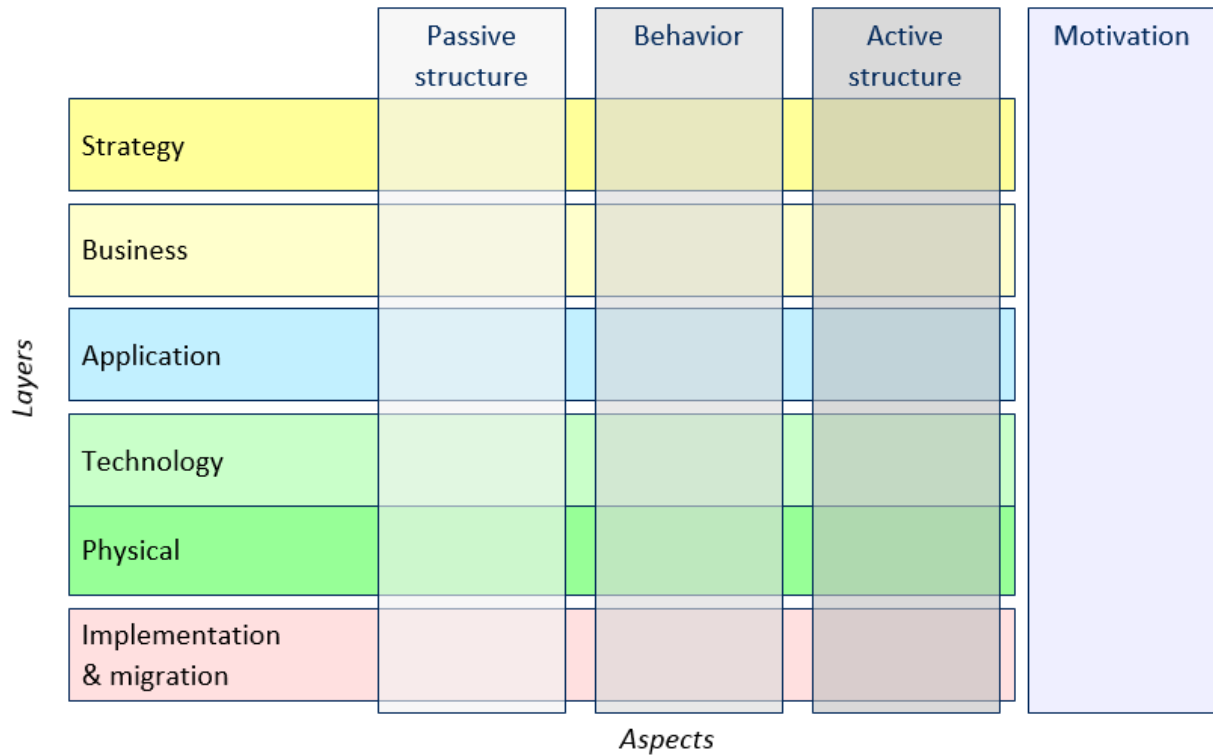


Figure 1) for organizing these concepts, a graphical notation for these concepts, and recommendations for viewpoints, which are visualization templates that address the concerns of particular stakeholders. The standard is public and free for end users. It can be extended through specialization of its concepts and relationships, and is supported by an increasing number of tools, consultancies and training organizations.

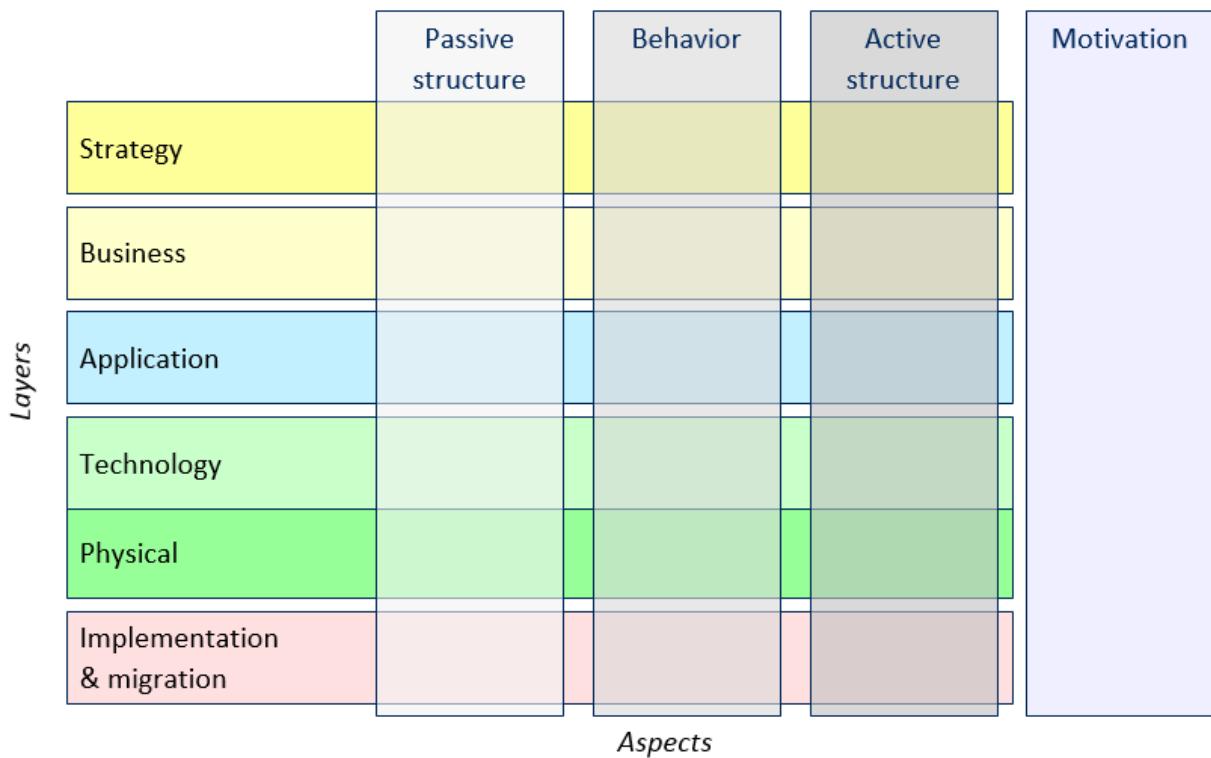


Figure 1. The ArchiMate 3 Framework ©2016 The Open Group [1]

Figure 2 illustrates how the components of the ArchiMate language support all phases of the TOGAF® [2] Architecture Development Method (ADM). In other words, as we will explore shortly, we can use ArchiMate models to fully describe enterprises and their transformations.

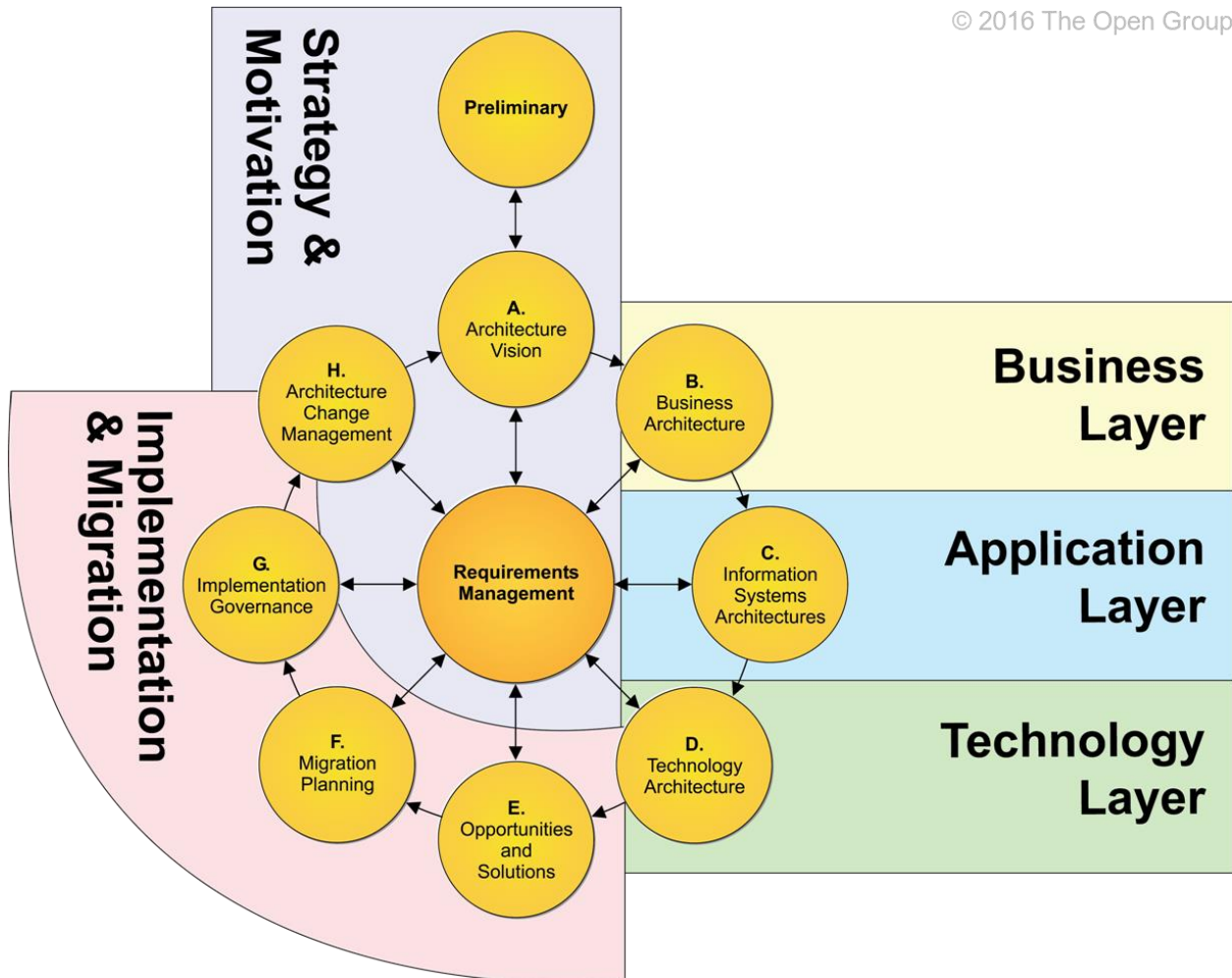


Figure 2. How the ArchiMate language supports all phases of the TOGAF ADM ©2016 The Open Group [1]

But What Can We Do with It?

We will now use a fictitious—but realistic—case study to describe what we can do with the ArchiMate language. Each of the next sections presents one or more views of an ArchiMate model that tells a story about the collection and analysis of Big Data to create business value. Big Data consists of datasets that cannot be handled efficiently with traditional centralized data architectures due to their extensive *volume, variety, velocity and variability*. These characteristics demand scalable architectures for efficient storage, manipulation and analysis [3].

So here is what we can do...

Depict an Industry Framework

The ArchiMate view in *Figure 3* provides an overview of the US National Institute Standards and Technology (NIST) Big Data Reference Architecture (NBDRA) [3]¹. It shows the relationships between three main groupings of capabilities, i.e. those provided by applications, infrastructure frameworks, and supporting fabrics. It also depicts the key roles that *realize*² or are *served by* these capabilities. The view also shows flows of information and software between capabilities.

¹ NIST has not, to the author's knowledge, published an ArchiMate representation of the standard. The ArchiMate models in this article have not, to the author's knowledge, been reviewed or approved by anyone representing NIST or its public working groups.

² Realization is a key relationship in the ArchiMate language. Section 5.1.4 of the specification states that "The realization relationship indicates that an entity plays a critical role in the creation, achievement, sustenance, or operation of a more abstract entity" [1].

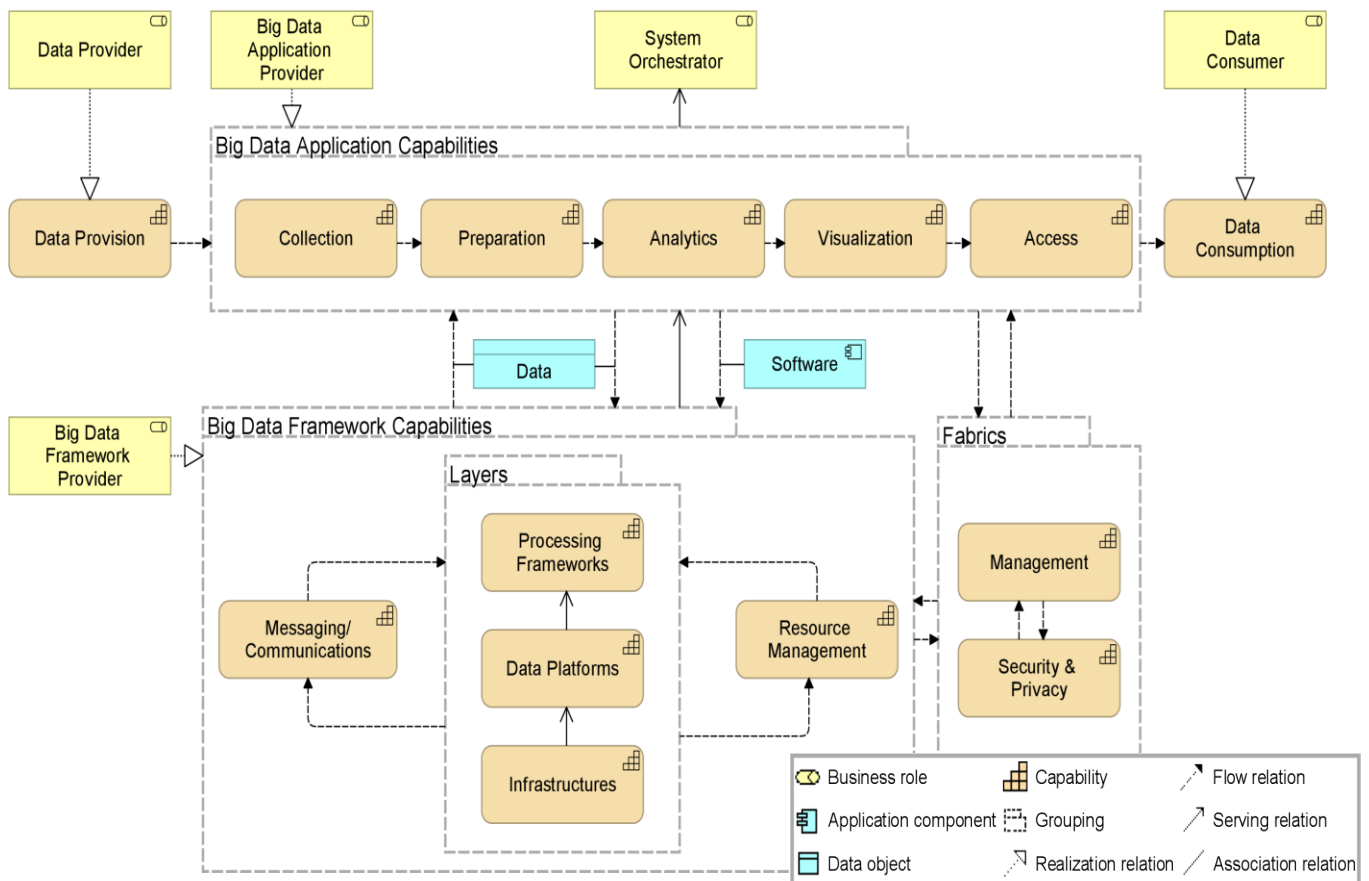


Figure 3. An Overview of the NIST Big Data Reference Architecture [3]

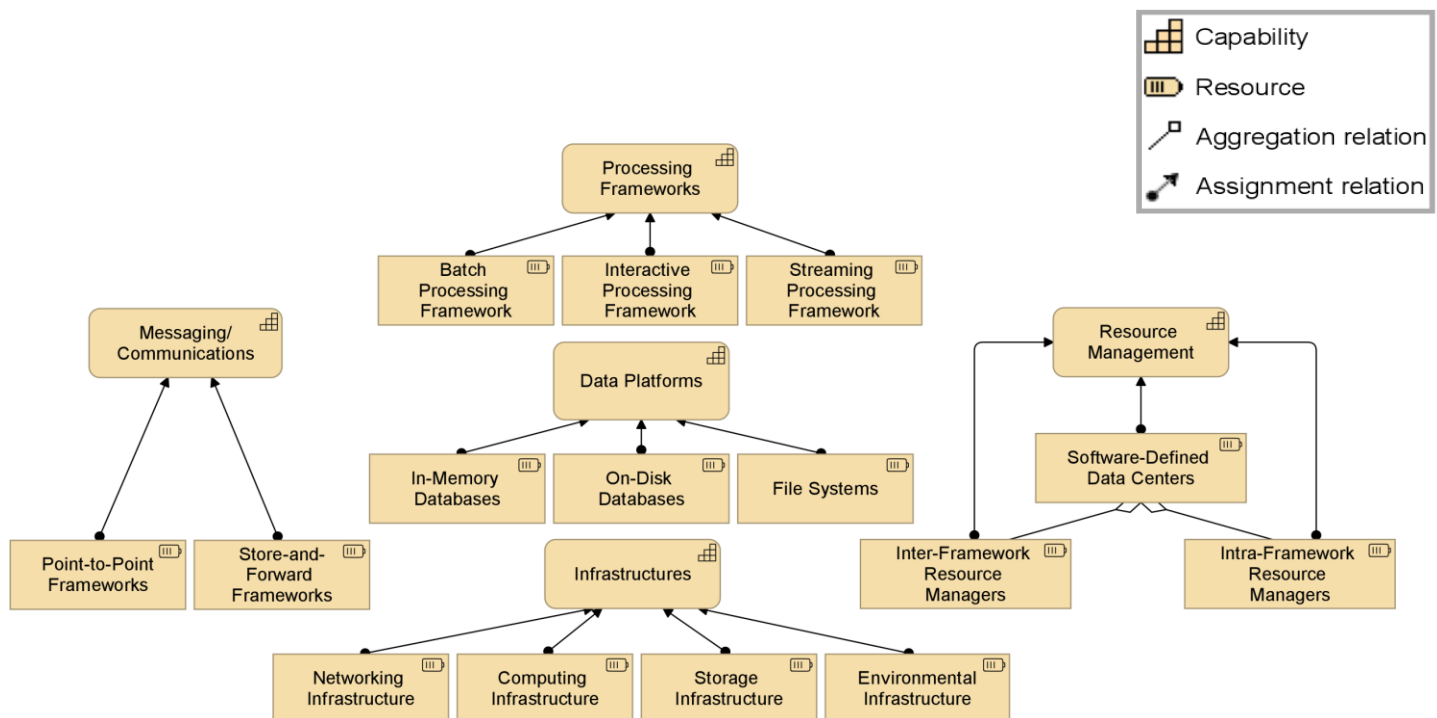


FIGURE 4. NBDRA BIG DATA FRAMEWORK CAPABILITIES WITH RESOURCES [3]

The view in Figure 4. NBDRA Big Data Framework Capabilities with Resources [3] explores in greater depth the Big Data Framework Capabilities introduced in Figure 3. It shows the resources that are used to deliver, or in ArchiMate terms, are *assigned to* each capability, and also how Software-Defined Data Centers *aggregate* two types of resource managers.

Show What's Driving an Organization

The view in Figure 5 shows the motivation of an enterprise, in this case a partnership between a health insurer and a digital health platform. The CEOs of both organizations share the goal of determining how tracked physical activity affects health, although they have different underlying motivations. The health platform CEO is eager to demonstrate the value of his company's offering, while the health insurer CEO wants to justify the cost of achieving another goal. The health insurer is providing fitness trackers to all new individual health plan members because of the CEO's concerns about the rate of growth of his company's individual line of business, and resulting individual claims experience, i.e. the types, frequencies and costs of claims.

In order to achieve their shared goal, the two CEOs must lead their organizations to satisfy four key requirements. They must collect claims data, collect fitness tracker data, link the two data streams by consumer and time period, and analyze the linked data. As seasoned leaders, they proceed to form the right team.

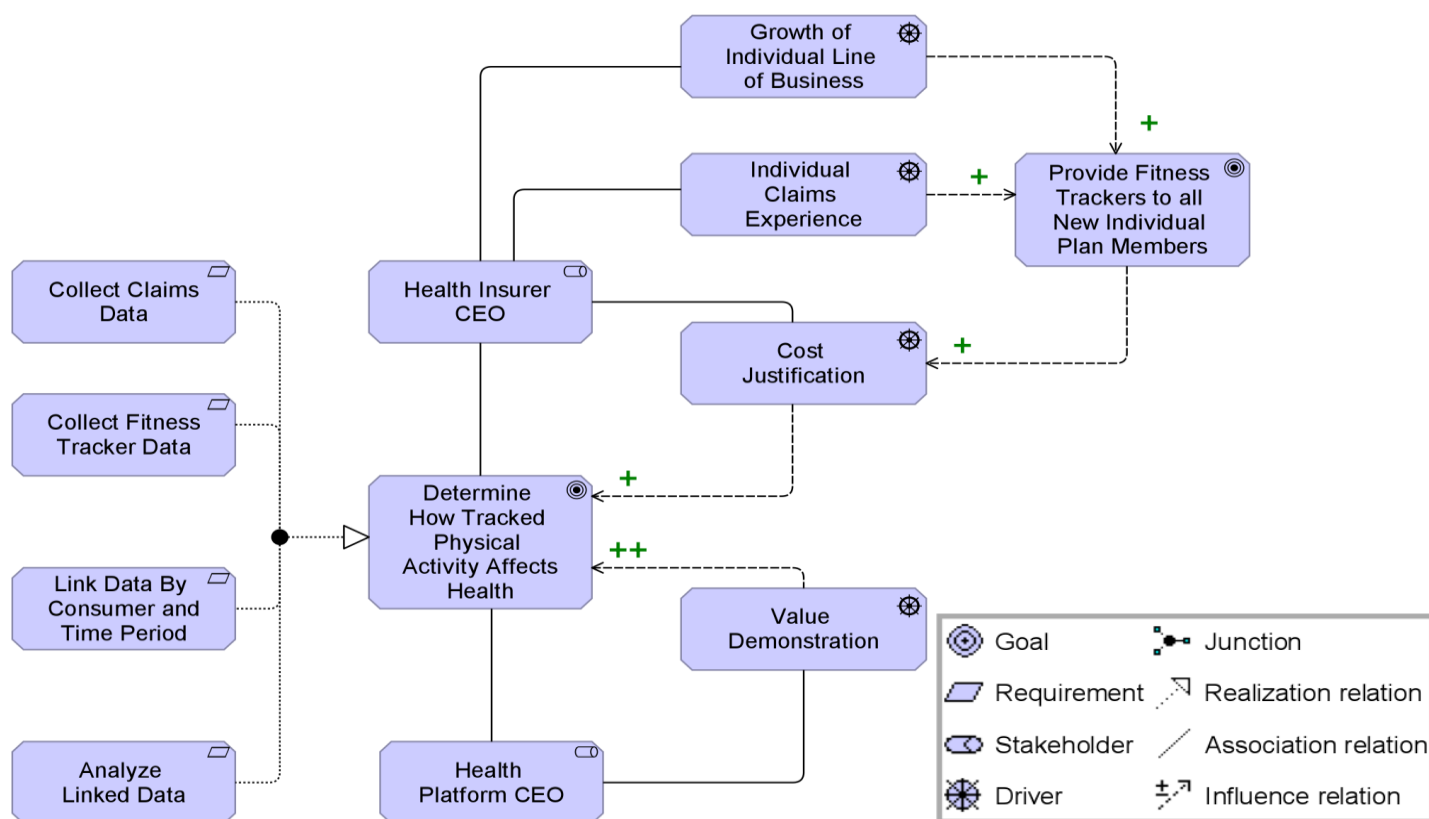


FIGURE 5. FITNESS TRACKER ANALYTICS MOTIVATIONS

Describe a Project Team

The Fitness Tracker Analytics project team (Figure 6) is led by a data scientist and a consulting physician from the health insurer, along with a big data solutions architect and a project manager from the health platform company. Together, these four fulfill the NBDRA role of System Orchestrator, while other people and organizations, i.e. ArchiMate business actors, fulfill other NBDRA roles. Both the health insurer and the health platform company are Data Providers, the health platform company is the Big Data Application Provider, and both an actuary from the health insurer and a market researcher from the health platform company are data consumers.

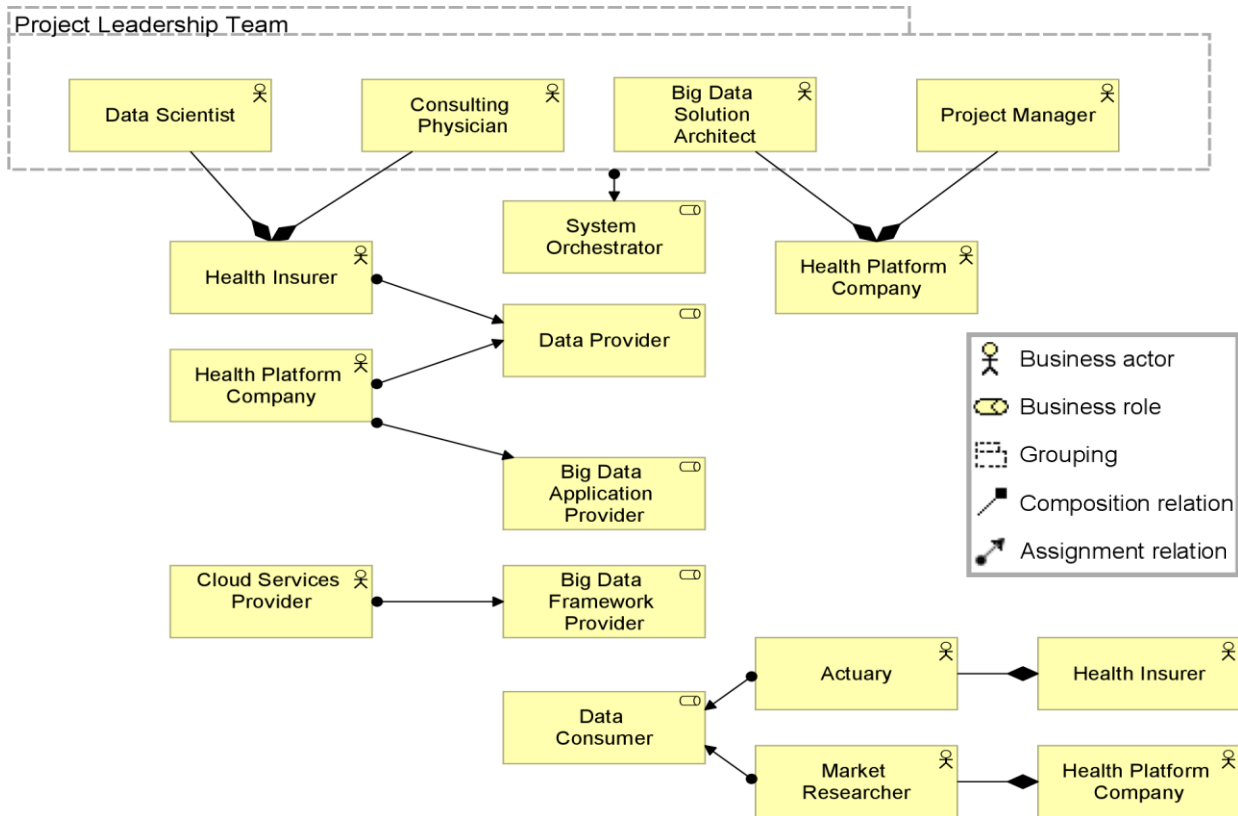


FIGURE 6. FITNESS TRACKER ANALYTICS PROJECT TEAM

Describe Applications and Relate Them to Organizational Strategy

Figure 7 shows the NBDRA Big Data Application capabilities and how they are realized by application behavior, specifically by a series of application processes. The Collection capability is realized by the Gather Claims and Activity Data process, which takes input data and places it in the Data Lake. Then the Link Data process, which realizes the Preparation capability, links together Activity, Claims, and Demographics data by associating each record with a uniquely identified person. The Analytics Capability is realized by the Compute Scores and Prepare Data for Visualization processes. Compute Scores is at the core of the entire effort, since it prepares a set of statistics for each consumer, including the degree to which tracked physical activity has improved the consumer's health.

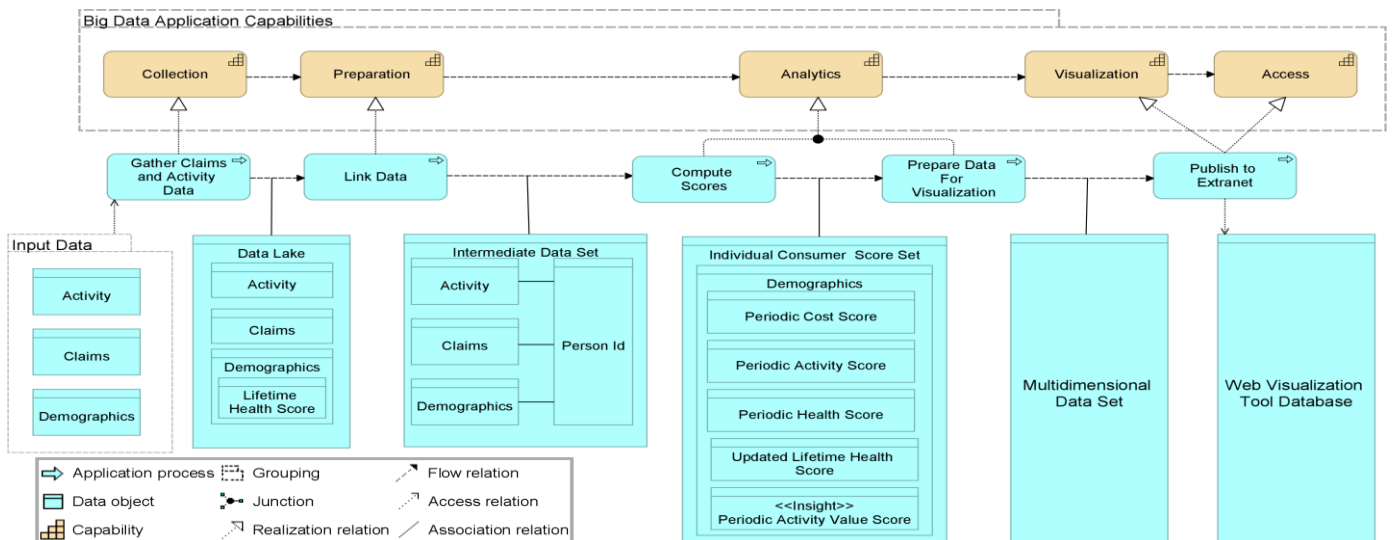


FIGURE 7. FITNESS TRACKER ANALYTICS DATA PROCESSING

Figure 8 examines a key part of the Compute Scores application process in detail. It contains a sub-process, Compute Periodic Claims Scores, which in turn contains three sub-processes that execute a MapReduce algorithm [4]. Beginning with Claims records that are linked to Person Ids³ and partitioned across multiple physical servers, a mapping process groups claims by Person Id within each partition. Then, a shuffling and sorting process groups claims by Person Id across all partitions, i.e. so each partition contains all the records of one or more consumers. Finally, a reducing process computes periodic claims scores for each consumer described in each partition. This results in periodic cost and health scores for each consumer sufficiently described by the input data set. All of these processes are executed by the Claims Analytics Script.

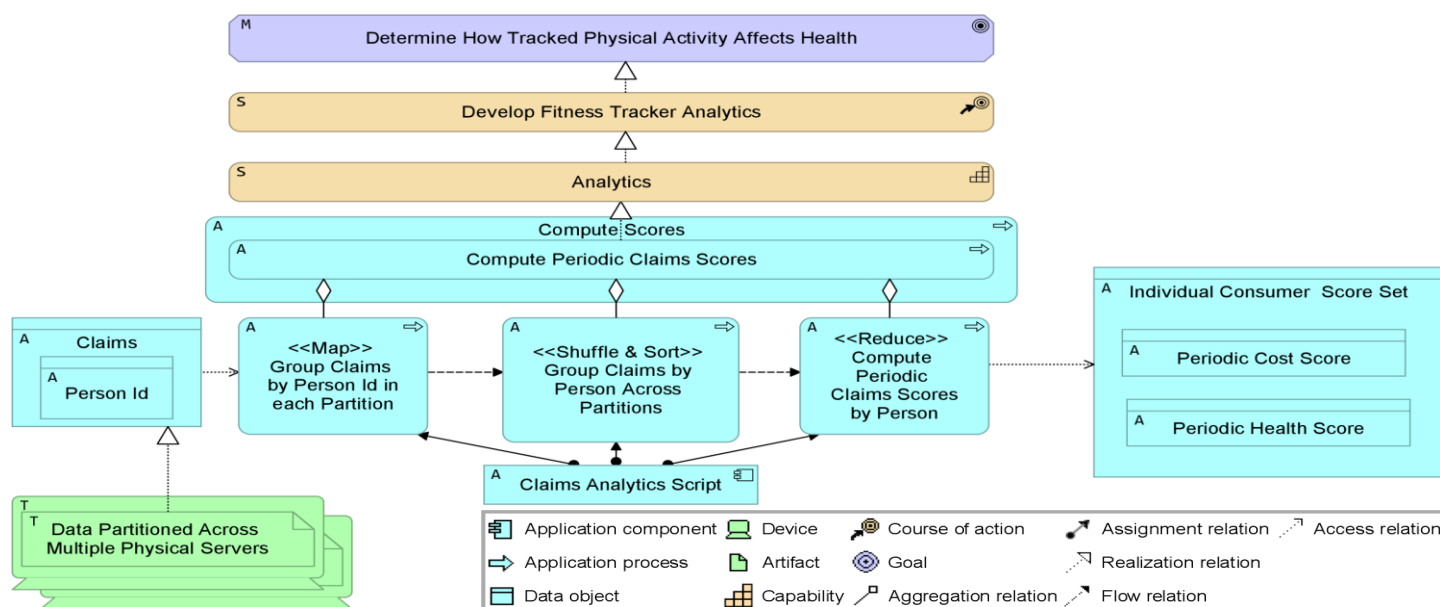


FIGURE 8. CLAIMS DATA PROCESSING WITH MAPREDUCE

Besides describing application behavior along with a bit of application structure and underlying technology, Figure 8 also relates that behavior to business strategy and motivation. The Compute Periodic Claims Scores application process realizes the Analytics capability, which in turn realizes the Develop Fitness Tracker Analytics course of action, i.e. the core strategy of this insurer-health platform partnership. This strategy realizes the shared goal of determining how tracked physical activity affects health.

Figure 9 describes the technology underlying the Claims Analytics Script, and ties the technology components back to the NBDRA by assigning them to the Big Data Framework resources described in Figure 4. The script uses a HADOOP stack [5]. Each system software component in the stack realizes either the Batch Processing Framework, the Intra-Framework Resource Managers, or the File Systems. The distributed processing environment realizes the Computing Infrastructure, while the data center housing that environment realizes the Environmental Infrastructure.

³ In this scenario, each Person Id uniquely identifies a consumer of both insurance from the health insurer and activity tracking services provided by the health platform via wearable devices.

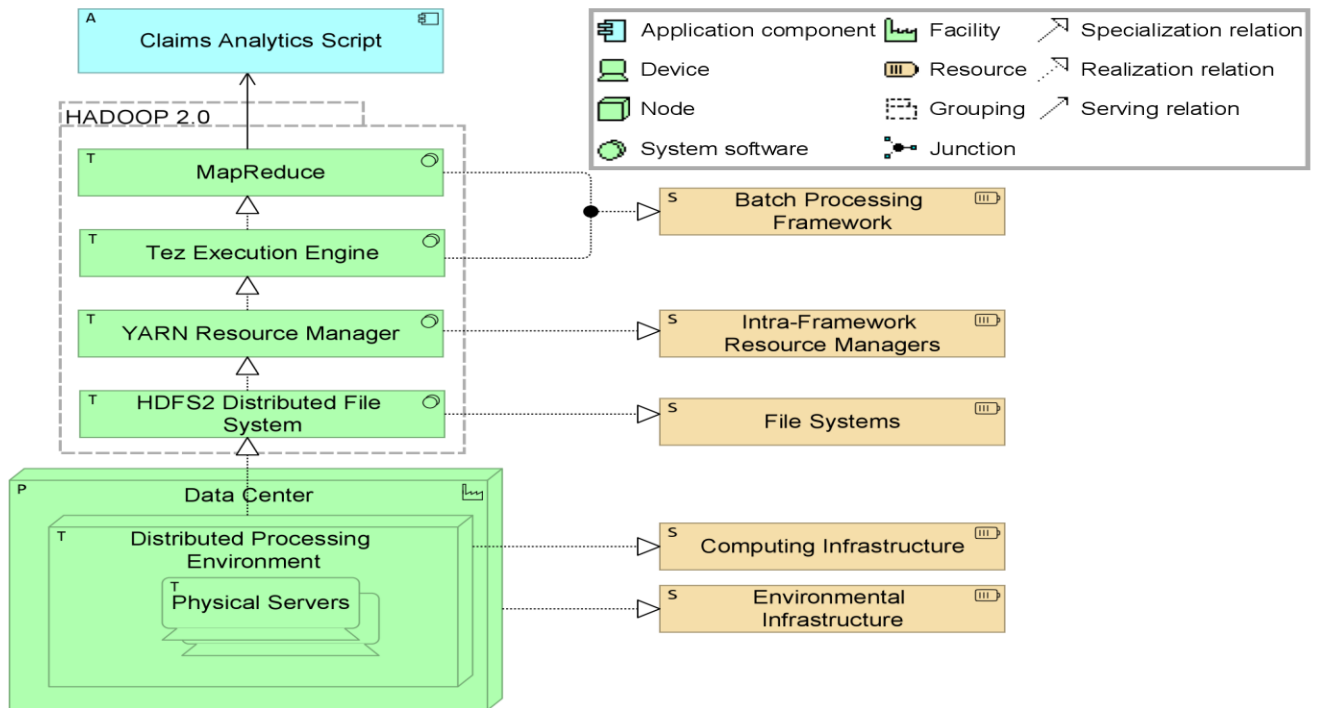


FIGURE 9. CLAIMS DATA PROCESSING TECHNOLOGY WITH RESOURCE REALIZATION

Depict a Multi-Phase Implementation Project

Figure 10 shows a series of plateaus⁴ realized by a corresponding series of project work packages. For example, after the project is approved, the Develop Initial ETL (Extract, Transform, Load) Processes work package realizes the Data Acquired plateau, in which the Gather Claims and Activity Data application process writes to the Data Lake.

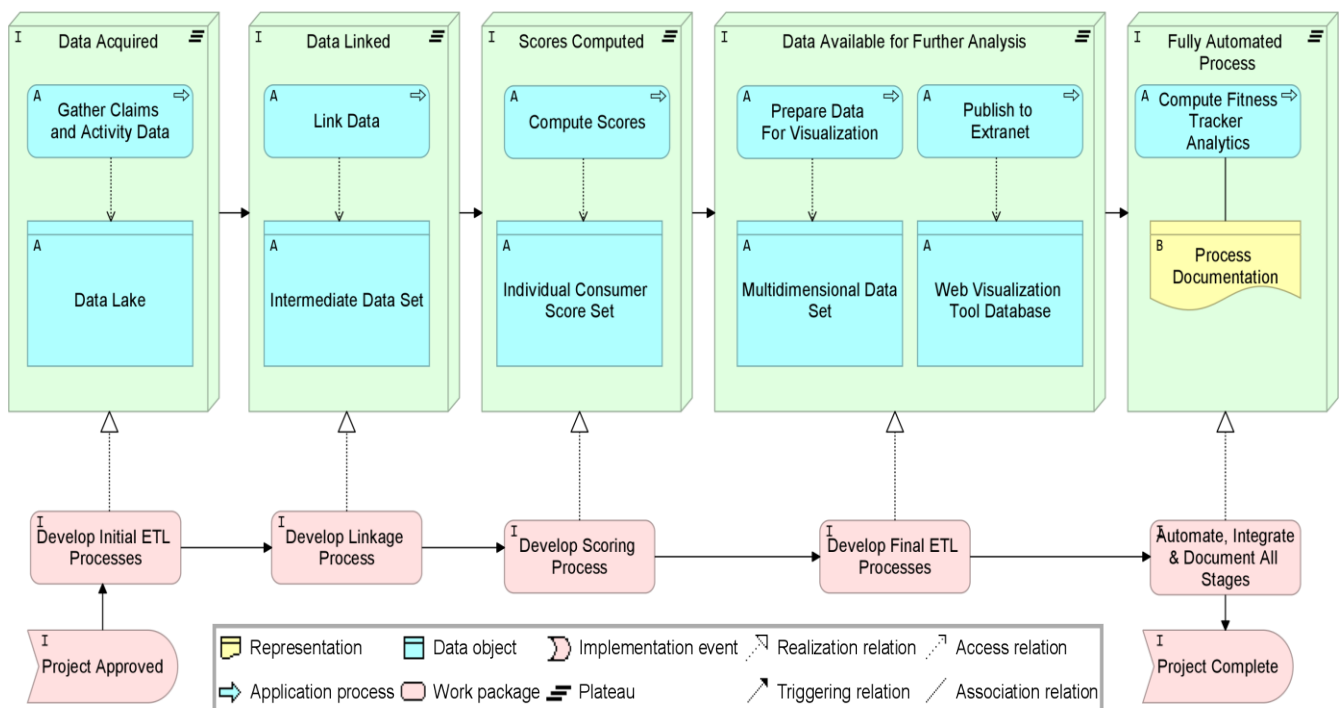


FIGURE 10. FITNESS TRACKER ANALYTICS IMPLEMENTATION PROJECT.

Model Order Fulfillment for a Physical Product

⁴ Section 13.2.4 of the ArchiMate 3.0 specification states that “A plateau represents a relatively stable state of the architecture that exists during a limited period of time” [1].

Figure 11 shows how ArchiMate models can span the physical and cyber worlds. A connected health device (CHD), such as a fitness tracker or a connected scale, is manufactured in Germany and sent to a distribution center in California via intermodal freight. A consumer uses a website hosted in a California data center to order a CHD. The order processing engine component underlying the website transmits the order securely over the Internet to a distribution center, where a pick list is printed and a worker uses it to pack and ship the order via overnight delivery to the consumer's mailbox.

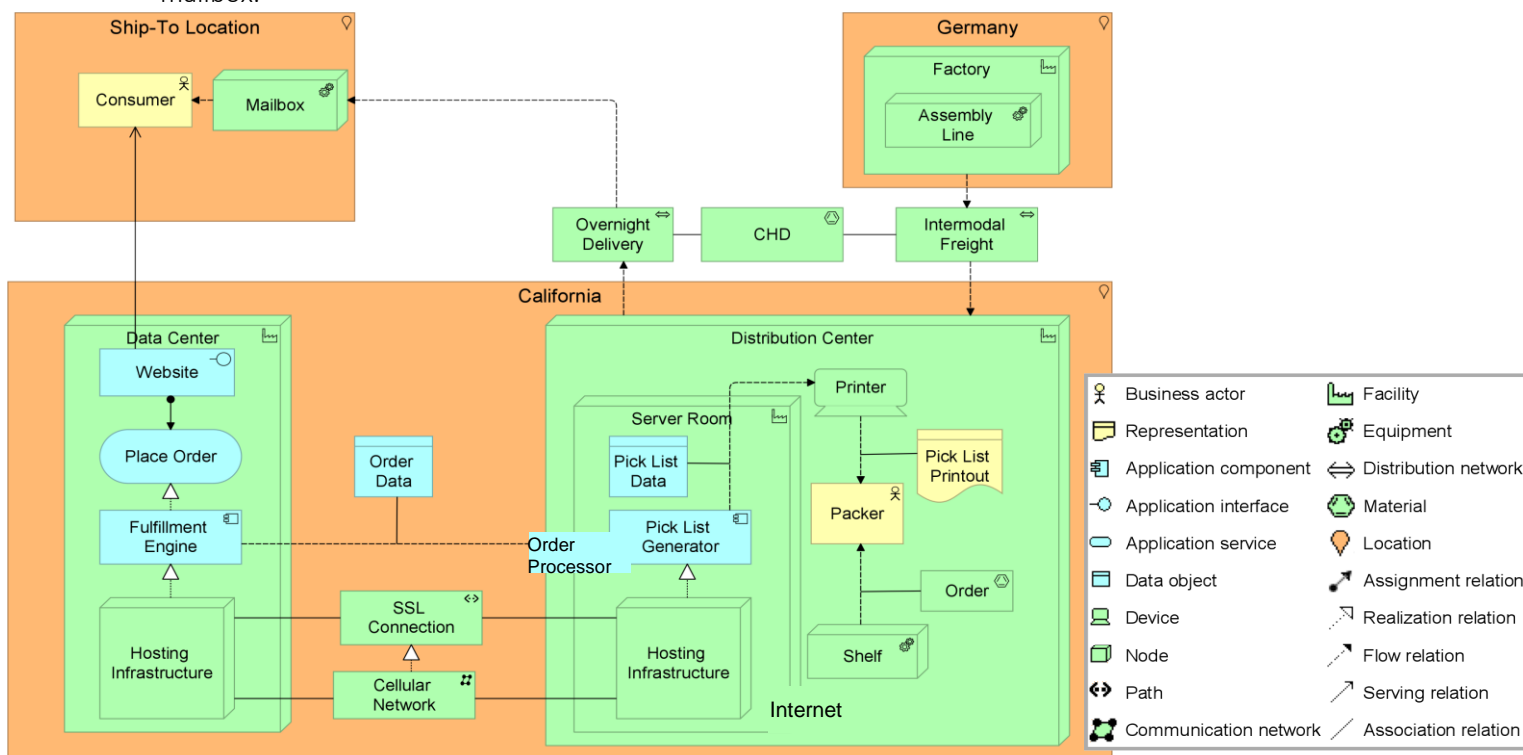


FIGURE 11. CONNECTED HEALTH DEVICE ORDER FULFILLMENT

Conclusion: What You Can Do with the ArchiMate Language

We have briefly touched on all major parts of the ArchiMate language (Figure 1), and have begun to see that the ArchiMate standard provides a complete language for describing enterprises, how they change, and the internal and external factors driving those changes. There is much more to the language, but hopefully this whirlwind tour has whetted your appetite to learn more.

You should now have a general sense of what the language can do. To learn more about the language, just start reading the specification [1], and download the free Archi tool [6], which is great for honing your newfound knowledge and putting it to good use.

About the Author

Iver Band is Chair of the Open Group ArchiMate Forum, an Enterprise Architect at Cambia Health Solutions in Portland, Oregon and a former editor of the Enterprise Architecture Professional Journal. He has over thirty years of technology experience as an engineer, architect and manager in enterprise IT and enterprise software R&D. For more about Iver, see his [LinkedIn profile](#) or his [SlideShare site](#).

References

The following materials are referenced in this article:

Please note that the links below function correctly at the time of writing but may not keep working.

- [1] ArchiMate® 3.0 Specification, an Open Group Standard (C162), published by The Open Group, June 2016; refer to: <http://pubs.opengroup.org/architecture/archimate3-doc/>.
- [2] TOGAF® Version 9.1, an Open Group Standard (G116), published by The Open Group, December 2011; refer to: <https://www.opengroup.org/togaf>.
- [3] NIST Big Data Interoperability Framework Version 1.0, published by the US National Institute of Standards and Technology, September 2015; refer to <https://www.nist.gov/el/cyber-physical-systems/big-data-pwg>.
- [4] MapReduce, published by Wikipedia contributors May 2017; refer to <https://en.wikipedia.org/wiki/MapReduce>.
- [5] What is Apache Hadoop®? published by the Apache Hadoop project, June 2017; refer to <https://hadoop.apache.org>.
- [6] Archi® version 4.02, software published June 2017 by Phil Beauvoir; refer to <https://www.archimatetool.com>.

CASE STUDY: ARCHIMATE MODELLING AT MAERSK

By Carl Chilley, A.P. Moller Transport & Logistics
and Marc Lankhorst, BiZZdesign

An Ongoing Journey of Transformation

Architecture Modelling at A.P. Moller - Maersk

Introduction

A.P. Moller - Maersk is an integrated transport & logistics company with multiple brands and is a global leader in container shipping and ports. Including a stand-alone Energy division, the company employs roughly 88,000 employees across operations in 130 countries. IT within the A.P. Moller group has been undergoing radical change in the last few years, stepping up to the digital challenge while seeking to maintain its market leading position. Architecture at all levels is seen as crucial to the success of this change, not just within IT but across the business as well. With a recent and ongoing organisation restructure to establish a new market-leading organisation— A.P. Moller Transport & Logistics (APM T&L) – sees the combination of six business units, driving home the need for a consistent, single view of what APM T&L is about, with architecture at the core of this change.

BiZZdesign is a software company that provides a model-based platform for collaborative business change called Enterprise Studio and used by APM T&L to support its modelling efforts. The company was founded in 2000 as a spin-off from the Telematica Instituut, a Dutch R&D institute that from 2002 to 2004 developed the first version of the ArchiMate modelling language discussed in this paper. BiZZdesign is very active in The Open Group's standardisation community around ArchiMate, having led the development of versions 2 and 3 of the language (The Open Group 2016).

This article describes APM L&T's ongoing architecture journey, its experiences with the ArchiMate language and BiZZdesign's software support, and the benefits that can be gained from using such models and tools. With this, we hope to inspire readers to employ similar approaches in their own organisations, in particular in industries outside the 'traditional' enterprise architecture users such as finance and government.

APM L&T's Architecture Journey

The Play Book

As with all large organisations, there are multiple factors to consider when defining the architectural journey that APM T&L is undertaking. The focus for a significant amount of the architectural change is with the "Play Book" project, central to the Enterprise Architecture Strategy team's definition of the way in which Architecture & Design is undertaken in APM T&L.

At the heart of the Play Book is the notion of *modelling*. This is driven by several critical success factors, of which the most important could be considered to be:

- *Consistency*. All solutions are modelled in the same way with the same underlying structures.
- *Readability*. Using a common graphical modelling language defines the vocabulary for the practitioners and readers alike.
- *Dis-ambiguity*. Given *consistency* and *readability*, dis-ambiguity follows as there should only ever be one way to interpret the solution being modelled.

- *Colocation*. All the models exist in one place, open for all to review and potentially...
- *Re-use*. Along with some logical partitioning of the repository to capture Architecture and Solution Building Blocks, Patterns and Reference Models, extant elements can be reused as part of a new solution.
- *Timeliness*. Because every element, relation and view is managed in the same place, re-use of existing models and parts will be to the current version and not to some older or decrepit variation.
- *Currency*. With the ultimate goal to move towards solutions modelled entirely in the modelling language and stored in the repository, there are no documents to be maintained (or not, as is often the case) as everything can be generated as required from the repository.
- *Integrity*. The sum of the previous factors underpins the overall integrity of the repository.

Modelling is controlled through the repository and associated tool. The tool captures the models from the architects and the repository, as would be expected, stores them, in this case in the cloud for full global access.

Another key factor is the clear separation of the space where the practitioners work and the space where other key stakeholders obtain information through reports, dashboards et cetera. This separation is essential to the integrity of the model space: no one other than trained practitioners can access the repository. Currently, this separation is achieved by publishing all reports on a centralised SharePoint site but this will be migrated to a portal solution in the near future.

Key Principles

To ensure that the Play Book is a success and used as expected, certain key principles have been defined to drive the work forward:

- *Full stakeholder engagement*. For the Play Book to be successful, the information it contains must not only be available but understandable by the non-technical stakeholders. Key design and delivery decisions will be driven by the models that underpin a solution, and it is critical that such models – or variants thereof – can be readily understood. We call these “technical” and “social” models.
- *Single place for the “truth”*. While there will always be master information sources that are consumed by the tool and the repository, the repository will provide links to such sources to ensure that all information can be found from the one place.
- *Just Enough, Just In Time*. All project-related work is subject to a project model of some sort, with well-defined stage gates and review points. Ensuring that there is sufficient material in the solution model for decisions to be made is not about maximising the detail but rather ensuring sufficient has been done. This is a balancing act but, when achieved, minimises the need for extensive rework when the inevitability of change impacts the project.
- *Risk-aware*. Early discovery of architecture and design risk is key to mitigation and to cost management: the longer the discovery time the more expensive the recovery. By putting risk and complexity discovery at core the heart of the architecture and design work from the earliest stages, mitigation can be identified early or avoided all together.
- *Full life cycle*. From an Enterprise perspective, architecture is a persistent need at all levels and across all phases of the delivery of a solution, including when in “live”. Too often, architecture is seen as an “up front” activity especially during the delivery phases. But for architecture to add value, it has to be a core discussion and decision-driving practice that transcends time.
- *Chose the appropriate language!* ArchiMate 3.0, outlined in Sect. 0, has been selected as the core modelling language for architecture and design. However, it has its limitations. ArchiMate is therefore the *reference framework* for all solutions, allowing other, more

suitable modelling languages to be used as required. These languages – BPMN, UML and DMN – add detail where required but in a context defined by the ArchiMate framework and linked through to it: the detail hangs off of the core solution elements defined in ArchiMate. This ability to relate to other languages is one of the key attributes of ArchiMate and was specifically designed into the language.

The ArchiMate Modelling Language

Enterprise architecture is an important instrument to support a company-wide integrated approach to development and change. It is a coherent whole of principles, methods and models that are used in the design and realisation of the enterprise's organisational structure, business processes, information systems, and infrastructure (Bernus et al., 2003). However, in practice these domains are often not approached in an integrated way. Every domain speaks its own language, draws its own models, and uses its own techniques and tools. Communication and decision making across domains is seriously impaired.

To create such an integrated perspective on enterprise architecture, you need proper techniques and tools that support the integrated description of the enterprise, creating a line of sight from strategy to realisation. Modelling languages are an essential instrument for the description and communication of architectures, and languages and tools have evolved more or less hand in hand. Many modelling languages, such as BPMN (Object Management Group 2013), UML (Object Management Group 2015) or DMN (Object Management Group 2016), provide concepts to model specific domains, e.g., business processes, software architectures or decision models, but rarely do they model the high-level relationships between these different domains. To this end, a language for modelling enterprise architectures is needed that focuses on inter-domain relations.

The ArchiMate language (Jonkers et al., 2003; Lankhorst et al., 2017; The Open Group 2016) provides exactly that. Initially defined in an applied R&D project, ArchiMate was adopted by The Open Group in 2009 and is now in version 3.0. The core of the ArchiMate language, the original ArchiMate 1.0 standard, covers the operational parts of the enterprise: the ubiquitous business, application and technology layers found in many architecture frameworks such as TOGAF (The Open Group, 2011). ArchiMate 2.0 extended this with concepts for describing the motivations behind architectural choices and concepts for modelling the architecture change and realisation process. ArchiMate 3.0 in turn added concepts for modelling the enterprise at a more strategic level, and for expressing physical technology in addition to the existing concepts for information technology. The figure below shows the coverage of ArchiMate 3.0 and its most common elements.

| | Passive structure | Behavior | Active structure | Motivation |
|----------------------------|-------------------|---|--|---|
| Strategy | resources | courses of action, capabilities | resources | stakeholders, drivers, goals, principles and requirements |
| Business | business objects | business services, functions and processes | business actors and roles | |
| Application | data objects | application services, functions and processes | application components and interfaces | |
| Technology | artifacts | technology services, functions and processes | devices, system software, communication networks | |
| Physical | material | | facilities, equipment, distribution networks | |
| Implementation & migration | deliverables | work packages | plateaus | |

Figure 12. ArchiMate 3.0 language coverage

To keep the language relatively easy to understand, there is a common structure underlying the concepts of each layer. For the language to be easy to learn and understand, each layer is based on the same structure. First, it distinguishes between the *structural* or *static* aspect and the *behavioural* or *dynamic* aspect. Structure elements are assigned to behaviour elements such as business processes, application functions and the like, to show who or what performs or is responsible for that behaviour. In addition to *active* structure elements such as business actors, application components and devices that display actual behaviour, i.e., the ‘subjects’ of activity, it also recognizes *passive* structure elements, i.e., the *objects* on which behaviour is performed.

Second, the language makes a distinction between an *external view* and an *internal view* on systems. When looking at the behavioural aspect, these views reflect the principles of service orientation. The *service* concept represents a unit of essential functionality that a system exposes to its environment. For the external users, only this external functionality, together with non-functional aspects such as the quality of service, costs etc., are relevant. Services are accessible through *interfaces*, which constitute the external view on the structural aspect. This language structure is shown in the next figure and you may recognize this in the examples that follow in the next sections.

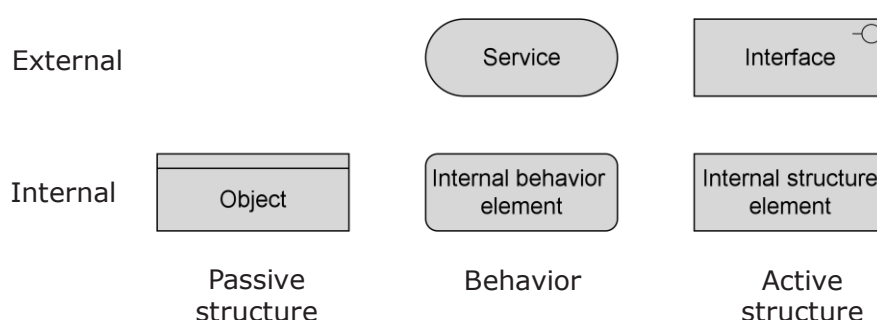


Figure 13. Core structure of ArchiMate

ArchiMate is also very strong and specific on relationships. It provides the following relationship types:

- *Composition* indicates that an element consists of a number of other elements.
- *Aggregation* indicates that an element groups a number of other elements.
- The *assignment* relation expresses the allocation of responsibility, performance of behaviour, or execution.
- *Access* models that behavioural elements can observe or act upon passive structure elements.
- The *servicing* relation models that an element offers its functionality to another element.
- *Realisation* indicates that an entity plays a critical role in the creation, achievement, sustenance, or operation of a more abstract entity.
- *Specialisation* indicates that an element is a particular kind of another element.
- The *Triggering* relation describes a temporal or causal relationship between elements.
- The *Flow* relation represents transfer from one element to another.
- *Association* models a relation between objects that is not covered by another, more specific relationship.

This precise specification of relationships allows you to perform various kinds of analyses that could not be done if every relationship was just a simple association. Thus, ArchiMate models can be used in answering questions such as:

- How important is this project to achieve our company goals?
- What is the potential impact of decommissioning this application?
- Who is responsible for these business processes?
- Where is my privacy-sensitive data used?
- What are potential performance bottlenecks in my infrastructure?

Another important aspect of ArchiMate is the separation between models and their notation, i.e., views on the architecture. The standard notation of ArchiMate is intended for architects and therefore rather 'technical' in nature, but it can also be denoted in different ways for other audiences, based on the same model structure. This is supported by the viewpoint mechanism outlined in the standard (The Open Group 2016).

Of course, this is just a small glimpse of what is possible with these models. APM L&T aims to use this analytical strength of ArchiMate in various ways, as is explained in the next sections. This article is too short to provide you with a full overview of the concepts, notation and use of the language. For more details, we refer you to the specification (The Open Group 2016) and the original book on the language, now in its fourth edition (Lankhorst et al., 2017).

Modelling at APM L&T

Making It Happen

Core to the use of the modelling approach in APM L&T is:

- The adoption of a core language as the reference framework outlined above;
- The adoption of specialised languages for specific and detailed models;

- A tool that is capable of supporting these languages while enhancing mixed language modelling and reporting;
- A repository to hold it all together.

As noted earlier, ArchiMate 3.0 was chosen as the core language, with ArchiMate experience gained in the architecture community over the last two years. In addition, the business community was using BPMN for modelling processes and the technical community was using UML for both high level and more detailed architecture and design. They were two obvious additional languages. With the move towards rules management and the need to handle events in the IT space, OMG’s Decision Model and Notation (DMN) was also selected as a specialised language.

The selection of the tool and repository was a joint consideration and BiZZdesign Enterprise Studio (BES) was selected, along with its Team Server software and platform. While the Team Server is managed by BiZZdesign in the Cloud, BES is installed on the modeller’s laptops, made necessary by the need for the modelling community to be able to work outside of the office and, on occasions, off line.

A training schedule has also been established, ensuring that all modellers were fully trained in both ArchiMate and the underlying tool and repository. This is critical and is handled in two waves, the first providing foundation training in the language and the tool and the second providing the APM T&L “how” for using the environment to deliver solutions. Foundation training provides access to a tool license for local use. The second wave of training then allows the modellers to be invited into the repository to work on specific projects and models.

A minimal *core model structure (CMS)* was also defined from which all solutions and building blocks would be derived. Because of the flexibility of the tool and the repository, the CMS is defined as an Enterprise Architecture construct comprising multiple models across multiple layers using multiple languages. Each of the layers has one or more models, with all models prescribing a recommended set of views that should be used to articulate the solution at this layer and its correspondence with the layer above. By using the CMS, common solution structures are imposed, which make the often complex solutions easier to navigate and to understand.

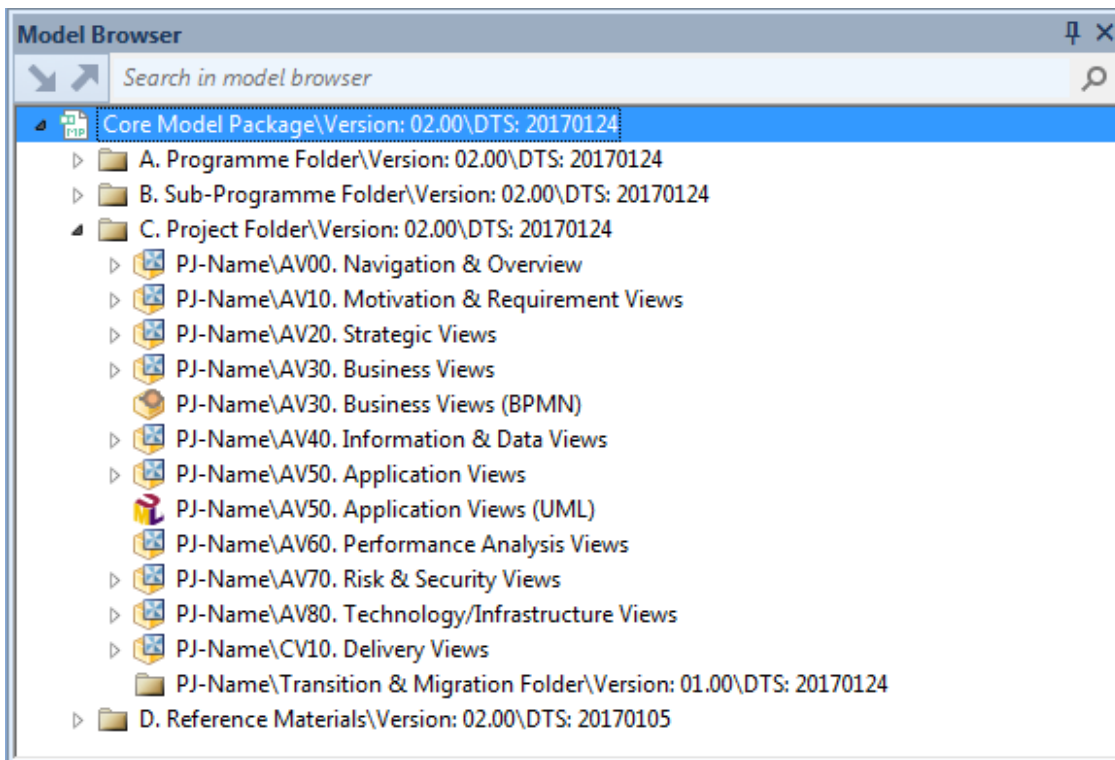


Figure 14. Core Model Structure, Shown in Project Form

Finally, the repository has a logical structure to it to segregate different projects and programmes while actively promoting the reuse of materials and techniques through the use of Architecture and Solution Building Blocks (ABB, SBB – realised through an *Application Catalogue*) and pattern libraries. Strict naming conventions apply to the models and their views within the repository, enabling modellers and other users to quickly ascertain which materials they need to work on or view.

In Reality

Theory is one thing, practice is another. Even with the training, structures and processes in place, there is still a lot of work to be done. Perhaps the most onerous example is the on-boarding of models into the repository (landing) and then their conversion into the common CMS. As noted earlier, ArchiMate had been in use for some time but without any formalised structures. Migrating these models – often from different toolsets and at different specification levels – requires a careful process, landing the old models into a *Transition & Migration Folder* (see *Figure 14*) prior to the repositioning and sometimes repurposing the models into the defined layers.

Even when on-boarded and re-aligned, there are additional considerations that underpin the successful analysis of the models and their views:

- **Making the architecture come alive for stakeholders.** Most stakeholders do not “speak” ArchiMate, hence the use of looser, “social” views. This is an ongoing area of research. As described in Sect. 0, ArchiMate supports the use of viewpoints for different audiences.
- **Right focus at the right time.** One of the most effective uses of modelling is to allow a single model view to express multiple aspects. As noted in the next bullet point, the breadth and depth of information you need in architecture and design varies with time: initial sketches and ideas do not require the same level of detail as later, more formed designs. Models can express these temporal and informational perspectives simply and effectively when both well thought through and when additional information overlays are used.
- **Early feedback on feasibility of solutions and trade-offs between alternatives.** Noting the previous point, there is often a need for sometimes speculative analysis at the early stages of a solution to choose between alternative solutions. Aside from the need for “broad brush” modelling, there is a pressing need for quantitative information upon which decisions can be made. The ability to both capture this information and overlay it onto models and views provides a simple and effective means to enable such decisions.
- **Complexity, risk and performance.** Following on from the previous three points, there is a continuous need to understand, quantify and trade between overall solution complexity, technical and system risk, and ability to meet performance requirements. Again, these can be clearly expressed with overlays onto models and views, as well as in the construction of quasi-technical (social) views that clearly illuminate and illustrate the competing factors.
- **Bridge the gap between architecture and operations.** It could be argued that “narrative + analysis = solution”, especially when the solution is augmented and overlaid with additional information to enable analysis. However, the narrative has to appeal to and be aligned with non-technical stakeholders, operations being a case in point but more generally anyone with an interest in the solution from the non-technical perspective. This is why the multiple layers of narrative that are present in modelled solutions have to be carefully defined and balanced depending on the reader. As noted earlier, this has given rise to the notion of “social Views”, where the rigour of the “technical view” is sacrificed for readability and clarity.

The models defined above can be used in many different ways. Of course, communicating intentions and results of APM L&T’s change efforts to various stakeholders is a primary purpose. Important in this respect is that a model needs to be accompanied by a story that explains it and puts it in the right context. Focusing on the right parts of a large enterprise model at the right time should also be part of that narrative: what is important, where do you need to look for answers?

A prime role of architecture models is also to provide early feedback on feasibility of solutions and trade-offs between alternatives. As everyone knows, fixing problems becomes much more costly in later stages of development. Well-structured models lend themselves to deeper analyses to assess this feasibility. In particular, one can use these models to uncover unnecessary complexity, unforeseen risks and potential performance problems. To that end, we are including architecture-level quantitative analyses of its IT applications and infrastructure to help assess issues such as scalability early on in the architecture and design process. It aims to use performance analysis methods similar to those explained by Lankhorst et al. (2017, Sect. 9.2) and Jonkers & Iacob (2006).

With solutions often traversing multiple layers (see *Figure 14*) and with multiple, often non-technical stakeholders being encouraged to make decisions based on the information present in the models, navigation through the model is critical. This is why the *AVOO: Navigation & Overview* model is paramount in the CMS. Good navigation provides a high-level narrative structure for the consumer of the model and different navigation paths are defined for different stakeholders and different concerns. BES allows for a very rich set of navigational structures to be defined, allowing different stakeholders to start their journey at different points (models) and be led through them in a manner that best meets their needs.

Moreover, such navigation should not be linear to be effective: it should allow the reader to journey along a directed path but also to venture “off piste”, able to explore other aspects of the solution while safe in the knowledge that she can return to their chosen path. Here, linear documents are far from ideal and the use of interactive HTML constructs allow for such navigation.

The Role of the Tool

As outlined above, APM L&T makes extensive use of modelling tools in its efforts. Of course, in any large organisation you will need more than simple drawing tools to manage architecture and design models. For the analyses of models mentioned above, this is even more important, since these analyses can only be done with model-based software tools. It was the very flexibility offered by BES that not only allowed us to utilise such powerful capabilities but to augment the power of the tool and therefore the strength of the underlying solution.

ArchiMate is a standard that actively supports customization of the language (The Open Group 2016, Chap. 15). Moreover, BES actively supports the capture of additional information (properties) against the ArchiMate elements and relations. The latter is easily handled through the use of *Profiles*. The captured properties can be manipulated and presented in a number of ways.

Profiles have allowed APM T&L to capture quantitative information against selected elements and relations such that a more complete picture of the solution can be analysed and presented. BES has a number of useful Profiles built in to the tool and APM L&T is actively extending this set to cover our own needs. Examples include:

- **Frequency of use and execution.** Applied to the “dynamic” relations of Access, Flow and Trigger, this defines how often something occurs as both an average and a peak. Displayed against one of the dynamic relations, this augments the operational understanding of the relation. It is also used as the basis for loads on networks (see “Payload” details).
- **“Payload” details.** A Payload is a specialisation of a Data Object that can be related to Flows, Accesses and Triggers. It captures both the size of the call and response, as well as the structure of the Payload. Displayed against its associated “Payload” this information overlay adds depth and detail to a model view. When used with frequency data, an analysis of the likely loads on a network can be undertaken.
- **Underlying transport protocols.** Applied to the “dynamic” relations of Access, Flow and Trigger, this captures the underlying transport protocol over which an interaction occurs. This is driven by an agreed set of such protocols that are modelled as ArchiMate Technology Path elements and allows for richer commentary in the Application layer. Displayed as labels or by colouring the relations appropriately, visual analysis and checking can be undertaken as to the

correctness of the protocol. When used with the Payload structure a visual check can reveal potential mismatches.

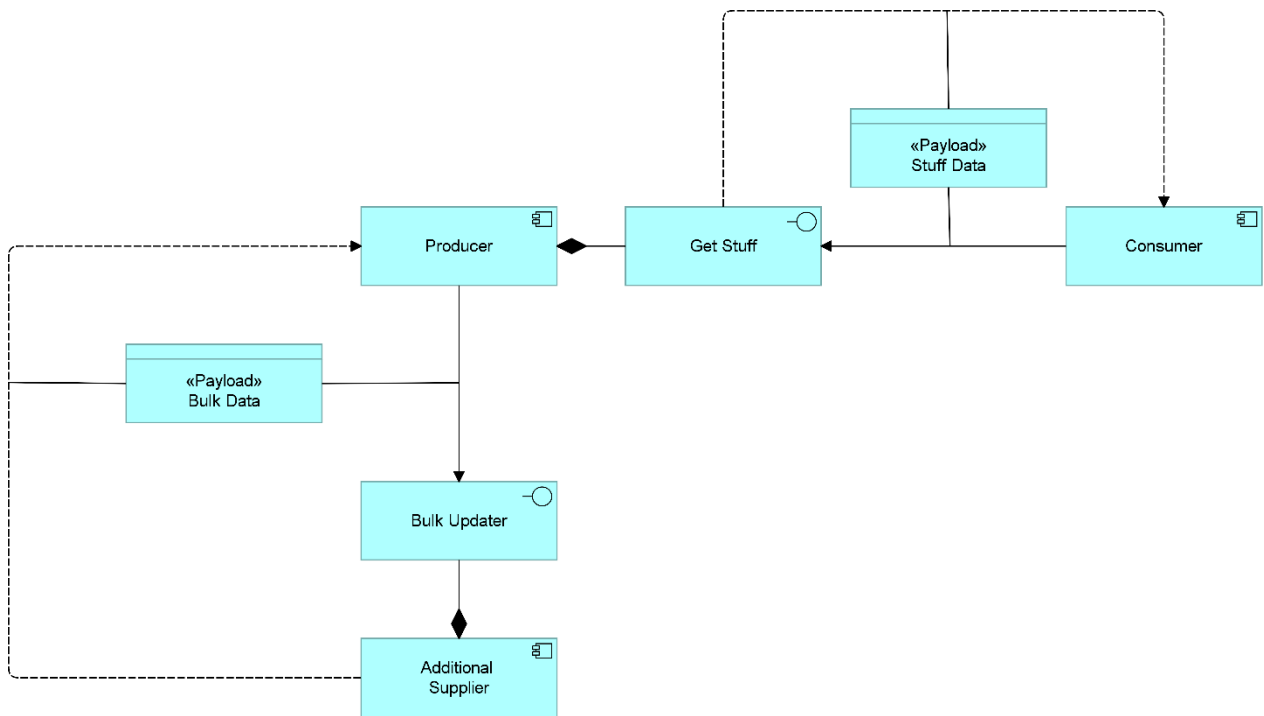


Figure 15 Vanilla View

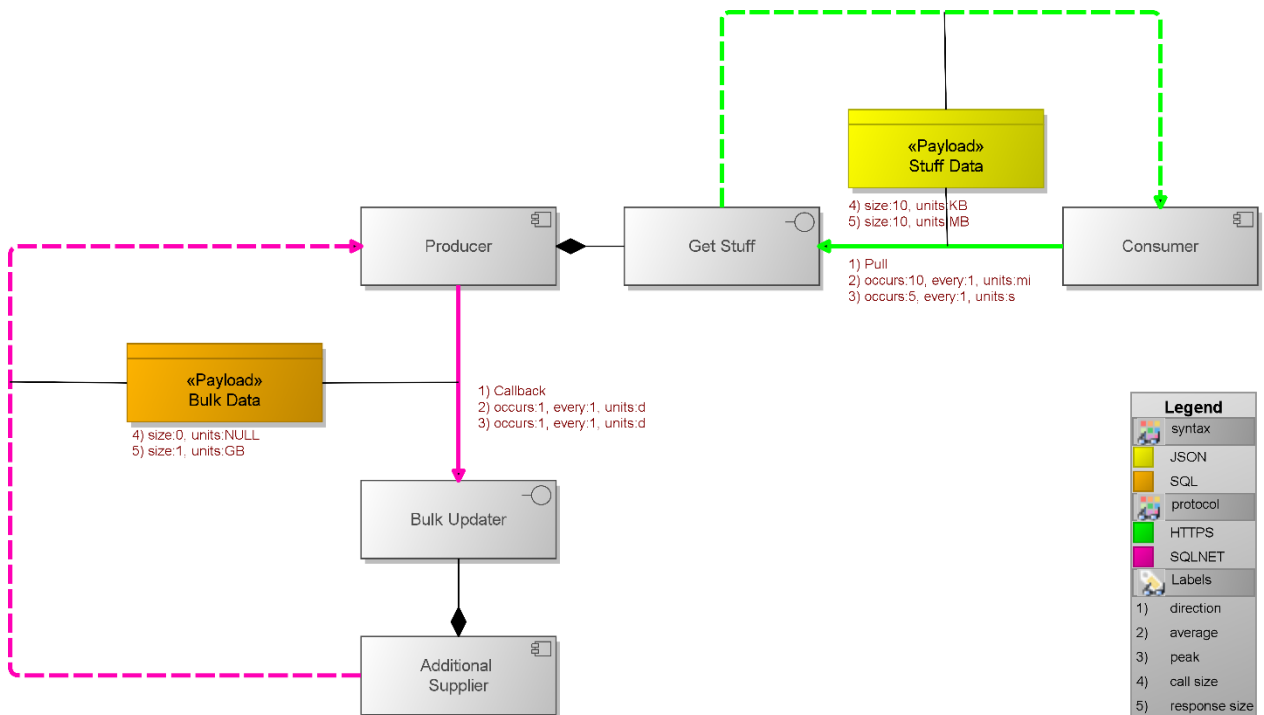


Figure 16 Animated View

The two figures above illustrate how a standard model view can be “animated” through the display of attributes defined in the profiles outlined above. While second figure could be seen as a little overloaded, it does add considerable detail to the vanilla view. Selective use of the tool’s rendering capabilities allows a single view to be used in multiple ways, presenting different data to different stakeholders. The animation also allows architects and designers to quickly outline potential areas of conflict and, as noted in the “Bulk Data” Payload element where the call size attribute has a ‘NULL’ value. NULL is used as the default value where selections have to be made, thereby forcing the modeller to be very specific in her intent.

The tool also allows for detailed interrogation of the models, both graphically and through enquiry, via a scripting language. Aside from allowing quantitative analysis to be undertaken within the tool, governance can be simplified by browsing the models, their elements and relations to check for consistency of use. APM L&T has defined an ArchiMate metamodel that details the minimal set of elements and relations that would be expected in any solution and across all layers. Using this as a backdrop, deviations from the metamodel can be easily detected, with in-built analysis tools available to handle a lot of this.

Import and export capabilities are also key. While stating the obvious there will always be a need to be able to export data to other tools for deeper analysis – e.g. quantitative data to an Excel spreadsheet – as well as generating catalogues of items of interest – e.g. Business Service catalogue; Business Process Catalogue; Application Interface catalogue. Equally, importing models into the repository as well as importing data mastered outside of the repository into the repository as ArchiMate elements – e.g. requirements from a requirements management system imported as ArchiMate Requirements – for analysis and mapping.

Finally – and as alluded to earlier – reporting is at the heart of the ability to communicate solutions to all of the interested stakeholders. While it is possible to generate detailed model- and language-specific reports in RTF and HTML, the real need are reports that can cover all models and all modelling languages, as used in the CMS. Making these reports interactive and accessible in a controlled and role-based manner ensures that the stakeholders get what they need in a form that is comprehensible to them. The tool offers all of this.

Results and Next Steps

Where Are We Now?

At the time of writing, multiple projects are active in the repository, looking at solutions at multiple levels and covering the breadth and depth of the scope articulated in the CMS. An initial set of profiles has been defined and is in use by the projects, and the use of interactive HTML reporting is taking hold.

But these are relatively early days and new projects are being on-boarded, new profiles being proposed, prototyped and considered and more practitioners trained at the basic and advanced levels. The use of modelling is being adopted as the norm and not the exception.

What's Next?

Aside from the continued growth outlined in the previous section, there are four key areas of growth that are vital to the success of the endeavour:

1. **More refined and targeted reporting.** Reporting is still very much at a project level at the time of writing and while there is some use of social views, it is in its infancy. With the advent of the reporting portal common, stakeholder-specific reports and dashboards will become commonplace. This will be underpinned by the ongoing research into the generation – potentially automatic – of social views and the increased breadth and depth of profiles and other APM L&T-specific specialisations.
2. **Extended use of information overlays.** A logical extension to the first point and effectively underpinning it, the development of profiles capturing additional properties provides the basis for a richer set of overlays to standard model views. While care is needed in their construction the portal offers report consumers finer grained control as to the information they can see. Augmenting the built-in overlays with consumer-selected perspectives, the consumer is offered more control as to the information that they can see and also manipulate. The early examples shown in the previous section are already proving to have a positive value for the modelling community, who will also drive the development of subsequent extensions.
3. **Transition away from documents: a single place for the truth.** One of the core targets for the modelling work, moving away from semi-moribund documents as the sole repository for

architecture and design information into a vital, emergent and continuously updated repository offers so many advantages, both to the practitioner and the consumer. As the repository captures more and more of the underlying IT and business landscape, there is a reduced need for searching around for information so often buried in often unknown spaces. Moreover, by enforcing an open disclosure policy, where project reports are generated on a regular basis and posted to the portal, folk get to see what is happening as it happens.

4. **Governance.** One of the most important aspect of the transition from documents to models is the way governance is undertaken: with the source (the model and the generated reports) and not through a proxy (documents that cut and paste ArchiMate, BPMN and UML diagrams from a number of sources). Because the model also houses the data (properties) critical to analysis, governance perspectives can be established to allow for reports and views to be generated that confirm (or otherwise) the validity and veracity of the model to the requirements it seeks to meet.

APM L&T is on a journey, one who's path and targets will change and transform over time. This is a natural state and one which, with the help of the community of practitioners and a dedicated term of modelling experts, will transform the way in which we execute architecture and design.

Advice to the Reader

Based on the experiences at APM L&T, we can make a number of recommendations to the reader.

First of all, ArchiMate is not a stand-alone language that aims to cover 'everything'. Rather, it works well in combination with more specialised languages for specific domains. UML for software, BPMN for processes, DMN for business rules, just pick and choose the languages that complement ArchiMate on the domains relevant to your organisation.

Second, when using a language like ArchiMate, you should define your own conventions of use. Any language has been designed with a broad spectrum of use cases in mind, but you will probably not need all of its features for your own situation. A logical organisation of your models such as APM L&T's core model structure will also help you keep things manageable.

Third, a good ArchiMate model can be used in many more ways than just to display views of the current or future state of your enterprise architecture. Already during the ArchiMate research project, various model analysis techniques were explored, and the examples given above show how useful such analyses are in providing guidance to the development of the enterprise, supporting decision-making and reducing the risk of failure by having early insights in the expected effects and properties of a proposed architecture. Chapter 9 of (Lankhorst et al., 2017) also gives many examples of such analyses, ranging from performance analyses like those mentioned above to impact of change analyses, security and risk evaluations, portfolio analyses, capability assessments, and more.

Finally, solid software support that goes beyond simple drawing tools is of course essential in making the most of your modelling efforts. Good tools enable you to quickly analyse your current situation and effectively test and plan changes. This simplifies decision making, lowers the risk of failure, and speeds up innovation.

In today's digital age, it is all about speed and effectively using IT to digitize your daily operations. Why not use this same power of software to digitize your change capabilities?

References

Bernus, P., Nemes, L., Schmidt, G. (2003). Handbook on Enterprise Architecture, Springer-Verlag, Berlin Heidelberg.

Jonkers, H. et al. (2003), Towards A Language For Coherent Enterprise Architecture Descriptions, In: Proc. 7th International IEEE Enterprise Distributed Object Computing Conference (EDOC 2003), September 16-19, 2003, Brisbane, Australia. IEEE Computer Society.

Jonkers, H., Iacob, M.-E. (2006). Quantitative Analysis of Enterprise Architectures. In: Konstantas D., Bourrières JP., Léonard M., Boudjlida N. (eds), Interoperability of Enterprise Software and Applications, Springer, London.

Lankhorst, M.M., et al. (2017). Enterprise Architecture at Work – Modelling, Communication and Analysis, Fourth Edition, Springer-Verlag, Berlin Heidelberg.

Object Management Group (2013). Business Process Modeling Notation (BPMN), Version 2.0.2 (formal/2013-12-09), Object Management Group. <http://www.omg.org/spec/BPMN/>.

Object Management Group (2015), OMG Unified Modeling Language™ (OMG UML) Version 2.5 (formal/2015-03-01), Object Management Group. <http://www.omg.org/spec/UML/>.

Object Management Group (2016a), Decision Model and Notation (DMN) Version 1.1 (formal/2016-06-01), Object Management Group. <http://www.omg.org/spec/DMN/>.

The Open Group (2011). TOGAF® Version 9.1, Van Haren Publishing. <http://www.opengroup.org/togaf>.

The Open Group (2016). ArchiMate® 3.0 Specification, Van Haren Publishing. <http://www.opengroup.org/archimate>.

Carl Chilley has been in IT for 40 years, starting in the engineering and scientific domain before moving into the systems and commercial domain. Carl is a “full width” Enterprise Architect with extensive experience in role and organisation modelling, process and service definition and delivery as well as IT, where he specialises in large-scale complex distributed systems. Carl has been the Chief Architect and Enterprise Architect on various programmes with large (> 30 projects) across the infrastructure, systems, applications and KID (Knowledge, Information & Data) domains with extensive involvement in organisation and process design.

Within APM T&L Carl is currently leading and shaping the way in which architecture and design is practiced and delivered, with modelling at the core. This is a multi-faceted approach, working from strategy through to infrastructure and technology, and is being adopted across many parts of the organisation without the need for enforced adoption.

Carl is a Chartered IT Professional (CITP) and a Member of the Institute of Engineering and Technology (MIET).

Marc Lankhorst is Managing Consultant and Chief Technology Evangelist at BiZZdesign. He is responsible for BiZZdesign's vision, market development, consulting and coaching on digital business design and enterprise architecture, and for spreading the word on the ArchiMate® language for enterprise architecture modelling, the Open Group standard of which he has managed the development. His expertise and interests range from enterprise and business architecture to service orientation, agile methods and digital business design. Marc has co-authored several books and numerous other publications, most notably ‘Enterprise Architecture at Work’, the original book on the ArchiMate language. He holds an MSc from the University of Twente and a PhD from the University of Groningen.

WEBINARS AND CONFERENCES

Here's a quick run-down of some of the EA-focused activity coming up around the globe:

ASSOCIATION OF ENTERPRISE ARCHITECTS & THE OPEN GROUP

As always, the AEA calendar is full of EA-related events. If you're an AEA member, remember to visit the <http://www.globalaea.org/> website regularly and check out the calendar of upcoming events. Here are a few you might be interested in:

The Open Group Ottawa 2017 | Making Standards Work® - e-Government

July 17th to July 20th

Ottawa, Canada

BCS - Enterprise Architecture Conference

July 17th

London, England

IT & Enterprise Architecture Forum, New Zealand

August 7th & 8th

Auckland, New Zealand

IRM UK Three One Day Workshops | Managing Your Information Asset

September 20th to September 22nd

London, England

The Open Group China 2017 | Becoming the Industry Benchmark through EA

September 21st & 22nd

Shenzhen, China

Don't forget, the AEA website also contains an archive of their webinars, which is available for members to view at any time. Check out the recent one from Len Feshkens entitled "The Siren Song of Linear Thinking".

IASA GLOBAL

Iasa Global run an ongoing program of in-person conferences (ITARC's) and virtual events (eSummits). Details of these events are constantly being updated as the events are organised, and participation as a speaker at these events is encouraged. You can find more about the upcoming ITARC's and eSummits by regularly visiting the following pages:

ITARC's

<https://www.iasaglobal.org/itarc-home/>

Planned events include London, Denver, Amsterdam and Austin, with more in the pipeline.

eSummits

<https://www.iasaglobal.org/esummit-home/>

The next event is August 23rd and focuses on the topic of Software Design for the Architect. Do you have something to share on the topic? Get in touch with Iasa and become a speaker. They will help guide you through the process.

If you have an event you want to let people know about, contact us at editor@eapj.org.

CALL FOR SUBMISSIONS

by Darryl Carr, EAPJ Editor

The Enterprise Architecture Professional Journal welcomes contributions in its fields of interest, which are enterprise, business, application, information, integration, technology and security architecture, as well as the strategic management of business and technology transformation. EAPJ publishes peer-reviewed material that advances its fields of interest and supports the careers of its readers.

EAPJ combines the strengths of peer-reviewed technical journals and professional newsmagazines. EAPJ invites submission of academic, feature, opinion, and interview articles. The editorial staff also considers other submissions, such as images, interactive graphics, video, and animations. Successful submissions contain actionable information that enhances the capabilities of professionals working within the EAPJ fields of interest.

Each issue consists of one or more main articles and one or more features, all centered on a theme introduced by the Editor's Welcome. Main articles are generally no more than 5,000 words in length, with body text interspersed with numerous callouts, graphics or tables.

EAPJ encourages submissions, readership and community participation from qualified individuals representing the widest possible variety of geographical regions, cultures, backgrounds and beliefs. Authors must properly attribute all referenced content and ensure that their submissions do not infringe upon any copyrights or intellectual property laws if published in the EAPJ. EAPJ encourages potential authors to contact the editor early on to receive guidance on developing material with the greatest likelihood of publication.

EAPJ also seeks expert reviewers to work with the editor and authors on developing and selecting main articles for the journal.

Please send expressions of interest, submissions, questions, ideas or comments to editor@eapj.org. Potential authors and reviewers should introduce themselves by describing their background briefly, supplying a resume or CV, or referencing an online profile.

You can also submit ideas for publication on our website. Visit <https://eapj.org> for details.