Foreword to the Third Edition

On January 31, 2012, The Open Group published version 2.0 of the ArchiMate® language for enterprise architecture modelling. This latest technical standard is now more aligned with TOGAF®, the world’s most popular enterprise architecture framework. This is an important milestone in the development of the profession, and this book, now in its third edition, provides much of the background and foundations of this development.

When Novay and its partners started the ArchiMate R&D project in 2002, they wanted to develop better means for communicating enterprise architectures. Until then, architects expressed their architectures either in proprietary tools and frameworks, with all the ensuing problems of vendor lock-in, or in fuzzy PowerPoint pictures that you could understand only if the architect was present to explain what all the boxes and lines meant. A well-founded open standard for architecture description was sorely needed.

Shortly after the project, consultants and educators began using it, the first commercial tools started to appear, and an active user community emerged. In 2008, The Open Group had just created a working group to establish a description language to complement TOGAF, when it was contacted by the ArchiMate Foundation. Since ArchiMate was already developed with TOGAF as one of its inputs, the match between the two created a great opportunity. In 2008, the ownership of ArchiMate was transferred to The Open Group and became a standard in 2009.

This proved to be an all-important step. With the rising popularity of TOGAF and the professional support of The Open Group, ArchiMate adoption figures have grown rapidly. At the time of writing, The Open Group’s ArchiMate Forum has some 70 member organisations, over 10 commercial and several open-source tools support the language, and its active LinkedIn group counts nearly 1700 members.

ArchiMate 2.0 provides a number of important extensions that make the fit between TOGAF and ArchiMate even closer. It improves collaboration through clearer understanding across multiple functions, including business executives, enterprise architects, systems analysts, software engineers, business process consultants and infrastructure engineers. The new standard enables the creation of fully integrated models of an organisation’s enterprise architecture, the motivation
behind it, and the programs, projects and migration paths to implement it. ArchiMate already follows terms defined in the TOGAF framework, and version 2.0 of the specification enables modelling through all phases of the TOGAF Architecture Development Method (ADM).

ArchiMate 2.0 provides enterprise architects with the tools and concepts necessary to create a consistent, integrated model that aligns more closely with TOGAF. It will increase interoperability and help enterprise architects establish a common language across the enterprise, raising the value and awareness of the discipline.

The growing use of models and standards is a sure sign of the maturation of any engineering discipline. This does not mean that enterprise architecture becomes a deterministic exercise, though. Rather, these instruments help managers and architects predict the effects of their actions, spot opportunities, and control risk, in the same way that navigational aids help a ship’s captain steer an optimal course in the prevailing currents and winds.

The Open Group
Reading, UK, February 2012

Allen Brown
President & CEO
Have you ever built a new house, or rebuilt an existing one? If you did, most likely an architect has been involved guiding you through the whole process of permits, drawings and construction. In this process, the architect creates insightful two- and three-dimensional drawings, models and views of the house. These show the structure of the house, its division into rooms (like the kitchen, living, bedrooms, and bathroom), its windows with views of the light, the networks of electricity, gas and plumbing, etc. The architectural design process of a house is a well-established discipline, using internationally accepted standards for describing and visualising the design, and various ways to present the design and analyse and calculate the strength of the proposed construction. The architect is well trained in the design methods, the modelling language and certain supporting tools.

Building or rebuilding an organisation is a much more complex and challenging task. First of all because the steps one has to take in order to (re)build an organisation are not standardised. One could start by first (re)designing business processes, followed by the application (re)design. Or one could first design generic application services, followed by designing business processes on top of these. Since a few years, The Open Group Architectural Framework (TOGAF) defines a standard way to take these steps. This enables enterprise architects to (re)design an organisation and its supporting IT systems in a uniform and standard way. The release of the improved TOGAF 9 version in February 2009 will lead to an even more uniform and better way to do this.

Secondly, building an organisation is a complex and challenging task because of the multifarious dependencies within an organisation. Many (often unknown) dependencies exists between various domains, like strategy, products and services, business processes, organisational structure, applications, information management, and technical infrastructure. Besides a having good overview over these different domains, one needs to be aware of their interrelationships. Together, these form the enterprise architecture of the organisation. In many cases, different languages and concepts are used to describe each domain, with no support for describing and analyzing relationships to other domains.
Until recently, a uniform and easy to use language for modelling and visualising enterprise architectures was lacking. ArchiMate, the modelling language described in this book, fills in this gap. It provides instruments to support enterprise architects in describing, analyzing and visualising the relationships among domains in an unambiguous way. ArchiMate is supported by different tool vendors and service providers. Many organisations are using it already as their company standard for describing enterprise architecture and its value has been proven in practice!

Just like an architectural drawing in classical building architecture describes the various aspects of the construction and use of a building, ArchiMate offers a common language for describing the construction and operation of business processes, organisational structures, information flows, IT systems, and technical infrastructure. This insight helps stakeholders to design, assess, and communicate the consequences of decisions and changes within and between these business domains.

Moreover, ArchiMate is now The Open Group’s open and independent modelling language for enterprise architecture. The specification of ArchiMate 1.0 has been released by The Open Group in April 2009. You can expect an even greater uptake of this language now that it has become a standard. Moreover, the synergy with TOGAF will provide enterprise architects with a very powerful approach, supported by methods, modelling languages and tools. Because ArchiMate is an open standard, it facilitates (model) interoperability and exchange of best practices. It is not a proprietary language from one tool vendor or service provider.

This book is about ArchiMate. It explains the background and the results of the research project that led to the realisation of the ArchiMate language. It also contains a description of the ArchiMate language itself, and many examples of its use for modelling, visualising and analysing enterprise architecture. The descriptions are based on the ArchiMate 1.0 specification published by The Open Group, and this second edition of the book adds more details on the relation between ArchiMate and TOGAF.

I cordially invite you to read this book. Reaching a second edition already proves its practical value. Convince yourself and start using ArchiMate!

---

Enschede, The Netherlands, February 2009

dr.ir. H.M. Franken, CEO, BiZZdesign
Chairman, ArchiMate Forum of The Open Group
‘Architecture’, in a broad sense, is the synergy of art and science in designing complex structures, such that functionality and complexity are controlled. The notion of architecture is used in a wide range of domains, from town planning to building and construction, and from computer hardware to information systems, each being characterised by the types of ‘structures’ or ‘systems’ being designed. However, we can recognise some common concerns in all these approaches.

To begin with, architecture, and hence the architect, is concerned with understanding and defining the relationship between the users of the system and the system being designed itself. Based on a thorough understanding of this relationship, the architect defines and refines the essence of the system, i.e., its structure, behaviour, and other properties.

This representation of the system’s essence, also called the ‘architecture’ of the system, forms the basis for analysis, optimisation, and validation and is the starting point for the further design, implementation, and construction of that system. The resulting artifacts, be they buildings or information systems, naturally have to conform to the original design criteria. The definition of the architecture is the input for verifying this.

During this process, the architect needs to communicate with all stakeholders of the system, ranging from clients and users to those who build and maintain the resulting system. The architect needs to balance all their needs and constraints to arrive at a feasible and acceptable design.

Fulfilling these needs confronts the methodology for defining and using architectures with demanding requirements. These can only be met if the architects have an appropriate way of specifying architectures and a set of design and structuring techniques at their disposal, supported by the right tools. In building and construction, such techniques and tools have a history over millennia. In information systems and enterprise architecture, though, they are just arising.

Important for an architecture description language is that the properties of the system can be represented in their bare essence without forcing the architect to include irrelevant detail. This means that the description language must be defined at the appropriate abstraction level.
If the architecture is concerned with the relationship between an enterprise and its IT support, the architect should be capable of expressing the structure, behaviour, and coherence of both the business processes and the IT support, such that one can use these specifications to get a thorough understanding of the architecture, to optimise it according to specific business goals, and to develop a strategy for introducing improvements in the current situation. This implies that the architecture description language should embrace easily understandable human notions of business processes and their IT support, far away from low-level implementation issues. It requires a level of comprehensibility of the description language by a broader audience than just the few specialists that are capable of understanding the obscurities of formal, mathematically oriented languages.

The very same applies to the methods that allow the architect to structure and manipulate architectural specifications such that their complexity can be controlled. Not in the least, the language and methods are the basis for unambiguous mutual understanding and successful collaboration between the stakeholders of the architecture. All stakeholders need to be aware about the implications of the choices in the architecture, and be capable of possibly influencing such choices.

This book presents the results of a research project that produced just that: a comprehensible, high-level design language for enterprise architecture, accompanied by a set of techniques and guidelines for visualisation and analysis of architectures. These results were validated in practice in real-life case studies in cooperation with several large, information-intensive organisations. Currently, various companies, ranging from vendors of architecture tools to consultants and other users of enterprise architecture, are implementing the results of the project.

This project is a prime example of the knowledge transfer for which the Telematica Instituut\(^1\) was founded. Both government and industry fund this Dutch national research institute. Its mission is to boost the innovative and competitive power of society by bridging the gap between academic research and its industrial application. The ArchiMate project, from which this book results, is a prime example of fruitful cooperation between these worlds. This proves the success of this knowledge transfer.

I hope and trust that the ArchiMate project not only proves to be an example of high-quality research in the important field of enterprise architecture, but also will have a considerable impact in practice.

---

\(^1\) In April 2009, the Telematica Instituut changed its name to Novay.
Many stakeholders within and outside the company can be identified, ranging from top-level management to software engineers. Each stakeholder requires specific information presented in an accessible way, to deal with the impact of such wide-ranging developments. To predict the effects of such developments and modifications of an organisation’s business and IT, it is necessary but very difficult to obtain an overview of these changes and their impact on each other, and to provide both decision makers and engineers implementing the changes with the information they need.

This book is about enterprise architecture, the practice that tries to describe and control an organisation’s structure, processes, applications, systems, and technology in such an integrated way. More specifically, we focus on methods and techniques for making and using integrated descriptions by means of architecture models, visualisation of these models for various stakeholders, and analysis of the impact of changes.

The unambiguous specification and description of components and especially their relationships in an architecture requires a coherent architecture modelling language. Such a language must enable integrated modelling of architectural domains and should be appreciated both by people from IT and by people with a business background. In this book, we present such an enterprise modelling language that captures the complexity of architectural domains and their relations and allows the construction of integrated enterprise architecture models. We provide architects with concrete instruments that may improve their architectural practice.

Furthermore, we provide techniques and heuristics for communicating with all relevant stakeholders about these architectures. Central to the communication of architectures is the notion of viewpoint. Viewpoints define abstractions on the set of models representing the enterprise architecture, each aimed at a particular type of stakeholder and addressing a particular set of concerns.

An architecture model is not just useful to provide insight into the current or future situation; it can also be used to evaluate the transition from ‘as is’ to ‘to be’. We therefore provide analysis methods for assessing both the qualitative impact of
changes to an architecture and quantitative aspects of architectures, such as performance and cost issues.

In order to make the approach we envisage practically feasible, architects require a tool environment, which supports the definition, generation, editing, visualisation, analysis, and management of architecture models and views. Moreover, such an environment should work in concert with existing domain-specific modelling tools, since we cannot expect architects to start using other tools, let alone other languages, than the ones they are used to. Although some tool developers are active in the enterprise architecture market, none currently provide a complete solution; some are focused on IT portfolio management, others on business process modelling, or on software architecture. We therefore present the design of a viewpoint-driven enterprise modelling environment that can provide just this support, and a vision on the future of model-driven enterprise architecture tooling. Currently, we are working with a number of commercial tool vendors to realise these ideas.

The modelling language and the other techniques in the book have been proven in practice in numerous real-life case studies. To put these instruments into context, the book also addresses the use of enterprise architecture models and techniques in governance, with a focus on alleviating the infamous business–IT alignment problem.
Audience

The intended audience of this book is twofold. On the one hand, we target enterprise, business, and IT architecture practitioners, especially those who are looking for better ways of describing, communicating, and analysing (enterprise) architectures. On the other hand, we aim for students of IT and (IT) management studying the field of enterprise architecture.
Overview of the Book

In the first chapter, we give an introduction to architecture in general and enterprise architecture in particular, outline its drivers, and describe the architecture process. Chapter 2 explains the methods and techniques currently used in this field. Following this, we outline the foundations of our approach to enterprise architecture modelling (Chap. 3). We then describe our view of architecture as being primarily a means of communication with all the stakeholders involved (Chap. 4).

Architectures are fruitfully used both in requirements analysis and design for new applications, business processes, etc., and to gain insight into existing systems (in the broad sense). In our approach, the use of architecture models has a central role; the modelling language used throughout the rest of the book is introduced in Chap. 5. Having a language is not enough: the architect also needs to be guided in its use, which is the topic of Chap. 6.

Many stakeholders with different goals or concerns in mind can view architectures. Each of these requires its own depictions of (part of) an architecture model, and the creation, use of such views and viewpoints is the topic of Chap. 7. Given that we have accurate models of an architecture, we can subject these models to various types of analysis, to establish for example what the impact of a change might be, or whether the performance of the technical infrastructure is sufficient given the applications and business processes that use it. These analyses are discussed in Chap. 8.

The practical applications of these modelling, visualisation, and analysis techniques are the topic of the next three chapters. In Chap. 9, experiences and best practices from case studies regarding the alignment of business, applications, and infrastructures are presented. These provide the context in which architectures are designed. Chapter 10 describes software tools that are currently available and our vision on and prototypes of future software support for enterprise architecture. Chapter 11 presents our practical experience with applying the techniques and prototypes in a number of real-life case studies. Finally, Chap. 12 provides a vision of the future: what is next; what comes ‘after’ architecture?
Acknowledgements

This book has resulted from the ArchiMate project, a Dutch research initiative that has developed concepts and techniques to support enterprise architects in the visualisation, communication, and analysis of integrated architectures. The ArchiMate consortium consisted of Telematica Instituut (now Novay), ABN AMRO, Stichting Pensioenfonds ABP, the Dutch Tax and Customs Administration, Ordina, Centrum voor Wiskunde en Informatica, Radboud Universiteit Nijmegen, and the Leiden Institute of Advanced Computer Science. Chapter 9 of this book results from the GRAAL project, a daughter project of ArchiMate. The GRAAL project was co-financed by the Telematica Instituut and the Centre for Telematics and Information Technology (CTIT) of the University of Twente, Enschede, The Netherlands.

Our special thanks go to Henk Jonkers for his help in editing the third edition of this book, to make it compliant with ArchiMate 2.0.

ArchiMate is now a trademark and a technical standard of The Open Group. More information on the ArchiMate standard can be found at http://www.archimate.org and http://www.opengroup.org/archimate.
## Contents

1 Introduction to Enterprise Architecture .......................... 1
   1.1 Architecture ..................................................... 1
   1.2 Enterprise Architecture ...................................... 3
   1.3 The Architecture Process .................................... 4
   1.4 Drivers for Enterprise Architecture ......................... 5
      1.4.1 Internal Drivers ......................................... 6
      1.4.2 External Drivers ....................................... 9
   1.5 Summary ...................................................... 10

2 State of the Art ................................................. 11
   2.1 Enterprise Architecture and Other Governance Instruments . . 11
      2.1.1 Strategic Management: Balanced Scorecard .............. 12
      2.1.2 Strategy Execution: EFQM ............................... 13
      2.1.3 Quality Management: ISO 9001 ........................... 15
      2.1.4 IT Governance: COBIT .................................. 15
      2.1.5 IT Service Delivery and Support: ITIL ................... 17
      2.1.6 IT Implementation: CMM and CMMI ..................... 18
   2.2 Methods and Frameworks ...................................... 19
      2.2.1 Enterprise Architecture Methods ....................... 19
      2.2.2 The IEEE 1471–2000/ISO/IEC 42010 Standard .......... 20
      2.2.3 The Zachman Framework .................................. 22
      2.2.4 The Open Group Architecture Framework (TOGAF) ...... 23
      2.2.5 OMG’s Model-Driven Architecture (MDA) ............... 26
      2.2.6 Other Frameworks ....................................... 27
   2.3 Description Languages ....................................... 29
      2.3.1 IDEF ....................................................... 30
      2.3.2 BPMN ..................................................... 31
      2.3.3 Testbed .................................................. 31
      2.3.4 ARIS ..................................................... 33
      2.3.5 Unified Modeling Language ............................... 34
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.6</td>
<td>Application Layer Concepts</td>
<td>.................</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>5.6.1 Application Structure Concepts</td>
<td>................</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>5.6.2 Application Behaviour Concepts</td>
<td>.................</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>5.6.3 Business–Application Alignment</td>
<td>................</td>
<td>93</td>
</tr>
<tr>
<td>5.7</td>
<td>Technology Layer Concepts</td>
<td>................</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>5.7.1 Technology Structure Concepts</td>
<td>................</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>5.7.2 Technology Behaviour Concepts</td>
<td>................</td>
<td>96</td>
</tr>
<tr>
<td></td>
<td>5.7.3 Application–Technology Alignment</td>
<td>...............</td>
<td>96</td>
</tr>
<tr>
<td>5.8</td>
<td>Relations</td>
<td>..........................</td>
<td>97</td>
</tr>
<tr>
<td>5.9</td>
<td>Motivation Extension Concepts</td>
<td>................</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>5.9.1 Stakeholder, Driver and Assessment</td>
<td>...........</td>
<td>102</td>
</tr>
<tr>
<td></td>
<td>5.9.2 Goal, Requirement, Constraint and Principle</td>
<td>...........</td>
<td>102</td>
</tr>
<tr>
<td></td>
<td>5.9.3 Motivation Extension Relations</td>
<td>................</td>
<td>104</td>
</tr>
<tr>
<td>5.10</td>
<td>Implementation &amp; Migration Extension Concepts</td>
<td>................</td>
<td>104</td>
</tr>
<tr>
<td></td>
<td>5.10.1 Implementation-Related Concepts</td>
<td>..............</td>
<td>104</td>
</tr>
<tr>
<td></td>
<td>5.10.2 Migration Planning Concepts</td>
<td>................</td>
<td>105</td>
</tr>
<tr>
<td>5.11</td>
<td>Language Extension Mechanisms</td>
<td>................</td>
<td>106</td>
</tr>
<tr>
<td></td>
<td>5.11.1 Adding Attributes to ArchiMate Concepts and Relations</td>
<td>...........</td>
<td>106</td>
</tr>
<tr>
<td></td>
<td>5.11.2 Specialisation of Concepts</td>
<td>................</td>
<td>107</td>
</tr>
<tr>
<td>5.12</td>
<td>ArchiMate and TOGAF</td>
<td>................</td>
<td>107</td>
</tr>
<tr>
<td>5.13</td>
<td>Modelling Example</td>
<td>................</td>
<td>109</td>
</tr>
<tr>
<td>5.14</td>
<td>Summary</td>
<td>..........................</td>
<td>113</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>..........................</td>
<td>115</td>
</tr>
<tr>
<td>6.2</td>
<td>The Modelling Process</td>
<td>................</td>
<td>116</td>
</tr>
<tr>
<td></td>
<td>6.2.1 Modelling as a Transformation Process</td>
<td>.............</td>
<td>117</td>
</tr>
<tr>
<td></td>
<td>6.2.2 Basic Modelling Activities</td>
<td>................</td>
<td>118</td>
</tr>
<tr>
<td></td>
<td>6.2.3 Types of Modelling Actions</td>
<td>................</td>
<td>120</td>
</tr>
<tr>
<td>6.3</td>
<td>Guidelines for Modelling</td>
<td>................</td>
<td>123</td>
</tr>
<tr>
<td></td>
<td>6.3.1 Before You Start</td>
<td>................</td>
<td>127</td>
</tr>
<tr>
<td></td>
<td>6.3.2 What to Capture in a Model?</td>
<td>.............</td>
<td>127</td>
</tr>
<tr>
<td></td>
<td>6.3.3 Modelling and Abstraction</td>
<td>................</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>6.3.4 Structuring Models and Visualisations</td>
<td>.............</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>6.3.5 Constructive Use of Modelling Breakdowns</td>
<td>.............</td>
<td>134</td>
</tr>
<tr>
<td>6.4</td>
<td>Readability and Usability of Models</td>
<td>................</td>
<td>137</td>
</tr>
<tr>
<td></td>
<td>6.4.1 Reducing the Visual Complexity of Models</td>
<td>.............</td>
<td>137</td>
</tr>
<tr>
<td></td>
<td>6.4.2 Representation Conventions</td>
<td>................</td>
<td>139</td>
</tr>
<tr>
<td>6.5</td>
<td>Summary</td>
<td>..........................</td>
<td>145</td>
</tr>
<tr>
<td>7.1</td>
<td>Architecture Viewpoints</td>
<td>..........................</td>
<td>147</td>
</tr>
<tr>
<td></td>
<td>7.1.1 Origin of Viewpoints</td>
<td>................</td>
<td>148</td>
</tr>
<tr>
<td></td>
<td>7.1.2 Architecture Viewpoints</td>
<td>................</td>
<td>149</td>
</tr>
<tr>
<td></td>
<td>7.1.3 Viewpoint Frameworks</td>
<td>................</td>
<td>150</td>
</tr>
</tbody>
</table>

7 Viewpoints and Visualisation | .......................... | 147  |
7.2 Models, Views, and Visualisations .............................................. 152
  7.2.1 Example: Process Illustrations ........................................... 154
  7.2.2 Example: Landscape Maps .................................................. 155
7.3 Visualisation and Interaction ................................................... 158
  7.3.1 Actions in Views .............................................................. 158
7.4 Creating, Selecting, and Using Viewpoints ..................................... 161
  7.4.1 Classification of Viewpoints .............................................. 161
  7.4.2 Guidelines for Using Viewpoints ........................................ 164
  7.4.3 Scoping ........................................................................... 164
  7.4.4 Creation of Views ............................................................. 165
  7.4.5 Validation ....................................................................... 166
  7.4.6 Obtaining Commitment ...................................................... 167
  7.4.7 Informing Stakeholders ..................................................... 168
7.5 Basic Design Viewpoints .......................................................... 168
  7.5.1 Introductory Viewpoint ...................................................... 171
  7.5.2 Organisation Viewpoint ..................................................... 171
  7.5.3 Actor Cooperation Viewpoint .............................................. 171
  7.5.4 Business Function Viewpoint ............................................. 173
  7.5.5 Product Viewpoint ............................................................ 176
  7.5.6 Service Realisation Viewpoint ........................................... 177
  7.5.7 Business Process Cooperation Viewpoint ................................ 177
  7.5.8 Business Process Viewpoint .............................................. 178
  7.5.9 Information Structure Viewpoint ....................................... 179
  7.5.10 Application Cooperation Viewpoint .................................. 180
  7.5.11 Application Usage Viewpoint ........................................... 181
  7.5.12 Application Behaviour Viewpoint ..................................... 183
  7.5.13 Application Structure Viewpoint ...................................... 184
  7.5.14 Infrastructure Viewpoint ................................................. 184
  7.5.15 Infrastructure Usage Viewpoint ....................................... 184
  7.5.16 Implementation & Deployment Viewpoint ................................ 186
7.6 Viewpoints for the ArchiMate Extensions ...................................... 186
7.7 ArchiMate and TOGAF Views .................................................... 187
7.8 Summary .................................................................................. 188

8 Architecture Analysis .................................................................. 189
  8.1 Analysis Techniques ............................................................... 189
  8.2 Quantitative Analysis ............................................................ 191
    8.2.1 Performance Views ............................................................. 192
    8.2.2 Performance Analysis Techniques for Architectures .............. 194
    8.2.3 Quantitative Modelling ...................................................... 196
    8.2.4 Quantitative Analysis Technique ....................................... 201
  8.3 Portfolio Analysis .................................................................... 206
  8.4 Functional Analysis .................................................................. 208
    8.4.1 Static Analysis ................................................................. 209
    8.4.2 Dynamic Analysis ............................................................. 211
  8.5 Summary ................................................................................ 219
9 Architecture Alignment ........................................ 221
  9.1 Introduction ............................................. 221
  9.2 The GRAAL Alignment Framework ....................... 222
    9.2.1 System Aspects .................................... 223
    9.2.2 The Aggregation Hierarchy ....................... 224
    9.2.3 The System Process ................................ 226
    9.2.4 Refinement Levels ................................ 226
    9.2.5 Comparison with Other Frameworks ................. 226
  9.3 Alignment Phenomena ..................................... 228
    9.3.1 Service Provisioning Layers ....................... 228
    9.3.2 Infrastructure Architecture ....................... 229
    9.3.3 Business System Architecture ..................... 232
    9.3.4 Strategic Misalignment ........................... 235
    9.3.5 Conway’s Law ..................................... 236
    9.3.6 The FMO Alignment Pattern ....................... 238
  9.4 The Architecture Process ................................ 238
    9.4.1 Methods .......................................... 239
    9.4.2 IT Governance ..................................... 240
  9.5 Summary .................................................. 242

10 Tool Support ................................................. 245
  10.1 Reasons for Enterprise Architecture Tooling .......... 245
  10.2 The Architecture Tool Landscape ....................... 246
  10.3 Tool Infrastructure ..................................... 247
  10.4 Workbench for Enterprise Architecture ................ 248
    10.4.1 Model Integration ................................ 249
    10.4.2 Viewpoint Definition ............................. 250
    10.4.3 Transparency and Extensibility ................... 251
    10.4.4 Software Architecture ............................ 251
    10.4.5 Exchange Formats ................................ 252
    10.4.6 Workbench at Work ............................... 252
  10.5 View Designer Tool ...................................... 254
    10.5.1 Viewpoint Rules for Creating Views and Visualisations ........................................... 255
    10.5.2 Defining Actions in Models and Views ............ 256
    10.5.3 Interactive Visualisation .......................... 258
    10.5.4 Example: The Landscape Map Tool ................. 259
    10.5.5 Comparison with Model–View–Controller Architecture ................................................. 261
  10.6 Impact-of-Change Analysis Tool ........................ 262
  10.7 Quantitative Analysis Tool ............................. 264
  10.8 Commercial Tool Support for ArchiMate ................ 265
  10.9 Summary .................................................. 266
11 Case Studies ................................................................. 269
  11.1 Process and Application Visualisation at ABP ................ 269
    11.1.1 ABP Meta-Model ............................................. 270
    11.1.2 Case Essentials ............................................. 271
    11.1.3 Concepts ..................................................... 272
    11.1.4 Viewpoints ................................................... 274
    11.1.5 Design of the Visualiser .................................. 277
    11.1.6 Case Study Results ....................................... 278
  11.2 Application Visualisation at ABN AMRO ..................... 279
    11.2.1 CITA Meta-Model ............................................ 280
    11.2.2 Case Essentials ............................................. 281
    11.2.3 Concepts ..................................................... 282
    11.2.4 Visualisation ................................................. 284
    11.2.5 Tool Design and Results .................................. 290
  11.3 Design and Analysis at the Dutch Tax and Customs
          Administration .................................................. 290
    11.3.1 Case Essentials ............................................. 291
    11.3.2 Views ......................................................... 291
    11.3.3 Performance Analysis ...................................... 298
    11.3.4 Case Study Results ....................................... 302
  11.4 Summary ........................................................... 302
12 Beyond Enterprise Architecture ....................................... 303
  12.1 The World Before Enterprise Architecture ................... 303
  12.2 The Advent of Enterprise Architecture ....................... 305
  12.3 The Extended Enterprise ....................................... 306
Appendix A – Language Meta-Model ..................................... 309
Appendix B – Graphical Notation ....................................... 313
Appendix C – Relations .................................................. 317
Trademarks .................................................................... 323
References .................................................................... 325
Index ........................................................................... 335
Contributors

F. Arbab  Centre for Mathematics and Computer Science (CWI), Amsterdam, The Netherlands; Leiden Institute of Advanced Computer Science (LIACS), Leiden, The Netherlands

S.F. Bekius  Dutch Tax and Customs Administration, The Hague, The Netherlands

M. Bonsangue  Leiden Institute of Advanced Computer Science (LIACS), Leiden, The Netherlands

H. Bosma  Ordina, Nieuwegein, The Netherlands

J. Campschroer  Ordina, Nieuwegein, The Netherlands

M.J. Cuvelier  Stichting Pensioenfonds ABP, Heerlen, The Netherlands

F.S. de Boer  Centre for Mathematics and Computer Science (CWI), Amsterdam, The Netherlands; Leiden Institute of Advanced Computer Science (LIACS), Leiden, The Netherlands

P. Fennema  BiZZdesign, Enschede, The Netherlands

L. Groenewegen  Leiden Institute of Advanced Computer Science (LIACS), Leiden, The Netherlands

S.J.B.A. Hoppenbrouwers  Radboud University, Nijmegen, The Netherlands

M.-E. Iacob  University of Twente, Enschede, The Netherlands

W.P.M. Janssen  Inzycht, Enschede, The Netherlands

H. Jonkers  BiZZdesign, Enschede, The Netherlands

D. Krukkert  TNO, The Netherlands

M.M. Lankhorst  Novay, Enschede, The Netherlands

P.G.M. Penders  Crescimento Universal, Buren, The Netherlands
H.A. Proper  Radboud University, Nijmegen, The Netherlands; CRP Henri Tudor, Luxembourg

D.A.C. Quartel  BiZZdesign, Enschede, The Netherlands

R.J. Slagter  GriDD, Enschede, The Netherlands

A.W. Stam  Almende, Rotterdam, The Netherlands

H.W.L. ter Doest  Dimpact, Enschede, The Netherlands

R. van Buuren  Novay, Enschede, The Netherlands

L. van der Torre  University of Luxembourg, Luxembourg

P.A.T. van Eck  University of Twente, Enschede, The Netherlands

D. van Leeuwen  BiZZdesign, Enschede, The Netherlands

G.E. Veldhuijzen van Zanten  Dutch Tax and Customs Administration, Apeldoorn, The Netherlands

R.J. Wieringa  University of Twente, Enschede, The Netherlands