THE DESIGN AND EVALUATION OF AN INTEGRATED ENTERPRISE ARCHITECTURE METAMODEL

by

Yzelle Roets

01227823

Submitted in fulfilment of the requirements for the degree

PhD Information Technology

in the

FACULTY OF ENGINEERING, BUILT ENVIRONMENT AND INFORMATION TECHNOLOGY

at the

UNIVERSITY OF PRETORIA

Supervisors:

Prof TJD Bothma

Dr JA Pretorius

August 2014

Declaration Regarding Plagiarism

I (full names & surname):	Yzelle Roets
Student number:	01227823
Topic:	The Design and Evaluation of an Integrated Enterprise Architecture Metamodel

Declare the following:

1. I understand what plagiarism entails and am aware of the University's policy in this regard.
2. I declare that this thesis is my own, original work. Where other people's work was used (whether from a printed source, the Internet or any other source) due acknowledgement was given and reference was made according to departmental requirements.
3. I have not used work previously produced by another student or any other person to hand in as my own.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of presenting it as his/her own work.

Signature

August 2014

Date

ACKNOWLEDGEMENTS

I give thanks to my heavenly Father for giving me the necessary talent, courage and strength.

> *"When I consider your heavens, the work of your fingers, the moon and the stars, which you have set in place, what is mankind that you are mindful of them, human beings that you care for them?"*
>
> Psalm 8:3-4

I want to express my appreciation and gratitude to:

- my husband, Gideon, for his patience and for providing me the space and opportunity to accomplish my dreams in more ways than one;
- my sons, Petrus and Henk, for their sincere interest and encouragement;
- my parents, Professor Niel and Doctor Marietjie Nelson, for their excellent example and carrying me in their prayers; and my mother for the final language editing;
- my supervisor, Professor Theo Bothma, for his excellent guidance in the academic processes and his positive encouragement;
- my co-supervisor, Doctor Jakkie Pretorius, for his constructive criticism, suggestions and support;
- the UP Library Service for providing access to resources, especially Reverend Danie Malan;
- the University of Pretoria for financial support; and
- the participants in the case studies, for their time, collaboration and assistance.

SOLI DEO GLORIA

THE DESIGN AND EVALUATION OF AN INTEGRATED ENTERPRISE ARCHITECTURE METAMODEL

SUMMARY

The research focuses on the relationship and interlinking between the different architectural domains within the enterprise architecture of an enterprise. The architectural domains are grouped together as business architecture, information architecture and technology architecture.

First, a literature review of the definitions, history, role, functions and qualities, existing frameworks, models and domains of enterprise architecture was conducted. The definitions; role and benefits; models, frameworks, ontologies and descriptive languages of the different architectural domains were then studied as background and basis. New definitions were deduced.

Second, the modelling methodology, process, elements and deliverables were investigated. An integration metamodel for enterprise architecture was developed, according to this modelling methodology. The metamodel is called the Relational Enterprise Architecture Metamodel or the REAM.

Third, the research methodology for the empirical research section was investigated and determined. The proposed conceptual metamodel has been assessed through a case study within three different industries each. The feedback from the case studies was used to modify/enhance the metamodel. The possibilities for application of the modified model were then assessed at the University of Pretoria.

The contribution of this research lies mainly in the development, empirical testing and refining of an integrated EA metamodel (REAM) as well as the development of new definitions for enterprise, business, information and technology architecture and relating them.

Keywords: enterprise architecture, integrated architecture model, information architecture, technology architecture, business architecture

DIE ONTWERP EN EVALUERING VAN ‘N GEÏNTEGREERDE ONDERNEMINGSARGITEKTUUR-METAMODEL

OPSOMMING

Hierdie navorsing fokus op die verhouding en koppeling tussen die verskillende argitektuurdomeine van ondernemingsargitektuur binne ‘n organisasie. Die argitektuurdomeine word saamgegroepeer onder besigheidsargitektuur, inligtings-argitektuur en tegnologie-argitektuur.

Eerstens, is ‘n literatuurstudie gedoen oor die definisies, geskiedenis, rol, funksies en eienskappe, bestaande raamwerke, modelle en argitektuurdomeine van onder-nemingsargitektuur. Die definisies, rol en voordele, modelle, raamwerke, ontologieë en beskrywende tale van die onderskeie argitektuurdomeine is bestudeer as agtergrond en basis. Nuwe definisies is afgelei.

Tweedens, is die modelleringsmetodologie, -proses, -elemente en aflewerbares ondersoek. ‘n Voorgestelde integrasiemodel vir ondernemingsargitektuur is hierna ontwikkel volgens hierdie modelleringsmetodologie. Die metamodel staan bekend as die *Relational Enterprise Architecture Metamodel* oftewel *REAM.*

Derdens, is die navorsingsmetodologie vir die empiriese navorsingsgedeelte ondersoek en bepaal. Die voorgestelde konseptuele metamodel is beoordeel deur een gevalle-studie elk in drie verskillende industrieë. Die terugvoer van die gevallestudies is gebruik om die metamodel aan te pas/te verbeter. Vervolgens is die toepassingsmoontlikhede van die aangepaste metamodel vir die Universiteit van Pretoria beoordeel.

Die bydrae van hierdie navorsing is hoofsaaklik geleë in die ontwikkeling, empiriese toetsing en verfyning van ‘n nuwe geïntegreerde onderneminsargitektuur-metamodel (die REAM) sowel as in die ontwikkeling van nuwe definisies vir ondernemingsargitektuur, besigheidsargitektuur, inligtingsargitektuur, tegnologie-argitektuur en die bou van verwantskappe tussen die argitektuurdomeine.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS III
SUMMARY IV
OPSOMMING V
TABLE OF CONTENTS VI
LIST OF TABLES XIII
LIST OF FIGURES XV
ABBREVIATIONS XXI
1 INTRODUCTION 1
1.1 THE RESEARCH PROBLEM 1
1.2 JUSTIFICATION AND RELEVANCY 4
1.3 LITERATURE REVIEW 5
1.3.1 Enterprise Architecture 10
1.3.2 Business Architecture 11
1.3.3 Information Architecture 12
1.3.4 Technology Architecture 13
1.3.5 Relationships between the Architectural Domains 14
1.4 RESEARCH METHODOLOGY 15
1.5 POINTS OF DEPARTURE 15
1.6 DEMARCATION OF RESEARCH 15
1.7 OVERVIEW OF CHAPTERS 16
SECTION A: LITERATURE STUDY 18
2 ENTERPRISE ARCHITECTURE 19
2.1 INTRODUCTION 19
2.2 DEFINITIONS OF ENTERPRISE ARCHITECTURE 20
2.2.1 Definitions in the Literature 21
2.2.2 Synthesised Definition 25
2.3 HISTORY OF THE ENTERPRISE ARCHITECTURE CONCEPT 26
2.3.1 History 26
2.3.2 Future 30
2.4 THE ROLE OF ENTERPRISE ARCHITECTURE 32
2.5 THE FUNCTIONS AND QUALITIES OF ENTERPRISE ARCHITECTURE 33
2.5.1 Aligning Domains 34
2.5.2 Provisioning Information 35
2.5.3 Driving Innovation/Change 37

2.5.4 Creating Agility ... 38
2.5.5 Improving Utilisation of Resources ... 39
2.5.6 Enhancing Processes ... 41
2.5.7 Reducing Risks ... 44
2.5.8 Linking with External Partners ... 44

2.6 DOCUMENTING ENTERPRISE ARCHITECTURE ... 45

2.6.1 The Benefits of Documenting Enterprise Architecture ... 45
2.6.2 The Challenges and Problems of Documenting Enterprise Architecture ... 48
2.6.3 The Impact of not Documenting Enterprise Architecture ... 50

2.7 EXISTING FRAMEWORKS ... 50

2.7.1 Zachman Framework ... 52
2.7.2 TOGAF ... 55
2.7.3 DoDAF ... 58
2.7.4 FEAF ... 60
2.7.5 Other Frameworks ... 63

2.8 EXISTING MODELS ... 66

2.8.1 Process Model ... 67
2.8.2 Relational Model ... 68
2.8.3 Causal Loop Diagram Model ... 69
2.8.4 Object-Oriented Model ... 70

2.9 LAYERS/DOMAINS OF ENTERPRISE ARCHITECTURE ... 70

2.10 SUMMARY AND CONCLUSION ... 73

2.10.1 Summary ... 73
2.10.2 Conclusion ... 78

3 BUSINESS ARCHITECTURE ... 79

3.1 INTRODUCTION ... 79

3.2 DEFINITIONS OF BUSINESS ARCHITECTURE ... 79

3.2.1 Definitions in the Literature ... 80
3.2.2 Synthesised Definition ... 84

3.3 THE ROLE AND BENEFITS OF BUSINESS ARCHITECTURE ... 84

3.4 EXISTING FRAMEWORKS AND MODELS ... 86

3.4.1 Business Components of Enterprise Architecture Frameworks ... 87
3.4.2 Business Architecture Frameworks ... 92
3.4.3 Modelling Languages ... 98
3.4.4 Ontologies ... 101

3.5 SUMMARY AND CONCLUSION ... 104

3.5.1 Summary ... 104
3.5.2 Conclusion ... 106

4 INFORMATION ARCHITECTURE ... 107

4.1 INTRODUCTION ... 107

4.2 DEFINITIONS OF INFORMATION ARCHITECTURE ... 107

4.2.1 Definitions in the Literature 108
4.2.2 Synthesised Definition....113
4.3 THE ROLE AND BENEFITS OF INFORMATION ARCHITECTURE 114
4.4 EXISTING FRAMEWORKS AND MODELS....116
4.4.1 Information Components of Enterprise Architecture Frameworks....116
4.4.2 Information Architecture Frameworks 121
4.4.3 Modelling Languages....129
4.4.4 Ontologies....131
4.5 SUMMARY AND CONCLUSION 132
4.5.1 Summary....132
4.5.2 Conclusion 134
5 TECHNOLOGY ARCHITECTURE....136
5.1 INTRODUCTION 136
5.2 DEFINITIONS OF TECHNOLOGY ARCHITECTURE 137
5.2.1 Definitions in the Literature 137
5.2.2 Synthesised Definition....141
5.3 THE ROLE AND BENEFITS OF TECHNOLOGY ARCHITECTURE 141
5.4 EXISTING FRAMEWORKS AND MODELS....143
5.4.1 Technology Components of Enterprise Architecture Frameworks 144
5.4.2 Technology Architecture Frameworks....151
5.4.3 Modelling Languages....156
5.4.4 Ontologies....157
5.5 SUMMARY AND CONCLUSION 158
5.5.1 Summary....158
5.5.2 Conclusion 159
6 RELATIONSHIPS BETWEEN THE ARCHITECTURAL DOMAINS....161
6.1 INTRODUCTION 161
6.2 DEPICTION AND TAXONOMY OF THE ARCHITECTURAL DOMAINS....163
6.2.1 Depiction of Enterprise Architecture....164
6.2.2 Taxonomy of Enterprise Architecture....168
6.2.3 A Proposed Integrated Depiction of Enterprise Architecture....170
6.3 DEFINING THE RELATION BETWEEN THE ARCHITECTURAL DOMAINS171
6.4 THE ROLE AND BENEFITS OF RELATING THE ARCHITECTURAL DOMAINS....176
6.5 EXISTING FRAMEWORKS AND MODELS....177
6.5.1 The Relational Properties of Enterprise Architecture Frameworks....178
6.5.2 Other Frameworks Illustrating Relationships....193
6.5.3 Models 208
6.6 SUMMARY AND CONCLUSION 209
6.6.1 Summary....209
6.6.2 Conclusion 213

SECTION B: DESIGN OF A RELATIONAL METAMODEL 214

7 THE MODELLING METHODOLOGY 215

7.1 INTRODUCTION 215

7.2 MODELLING PROCESS 215

7.2.1 Defining the Problem 216
7.2.2 Background Research 217
7.2.3 Design/Development 217
7.2.4 Evaluation and Refining 217
7.2.5 Completion 218

7.3 MODELLING ELEMENTS AND DELIVERABLES 218

7.3.1 Applicable Standards 219
7.3.2 Documentation 220
7.3.3 Design Principles 221
7.3.4 Definition of Terms 222
7.3.5 Depictions 222
7.3.6 Viewpoint Specifications 222
7.3.7 Architectural Views 223
7.3.8 Alignment Approach 224
7.3.9 Stakeholders and Concerns 224
7.3.10 Applying the Metamodel 224

7.4 SUMMARY AND CONCLUSION 225

7.4.1 Summary 225
7.4.2 Conclusion 226

8 THE RELATIONAL METAMODEL 227

8.1 INTRODUCTION 227

8.2 DESIGN/DEVELOPMENT STEP 227

8.2.1 Relevant Existing Frameworks 228
8.2.2 Considering Design Alternatives 229
8.2.3 Development of the Chosen Alternative 239

8.3 DESCRIPTION OF MODEL/FRAMEWORK 245

8.3.1 Applicable Standards 246
8.3.2 Documentation 247
8.3.3 Architecture Principles 251
8.3.4 Definition of Terms 257
8.3.5 Depictions 258
8.3.6 Viewpoint Specifications 261
8.3.7 Architectural Views 262
8.3.8 Alignment Approach 262
8.3.9 Stakeholders and Concerns 266
8.3.10 Applying the Metamodel 266

8.4 SUMMARY AND CONCLUSION 269

8.4.1 Summary 269
8.4.2 Conclusion 272

SECTION C: EMPIRICAL RESEARCH – CASE STUDIES 274

9 RESEARCH METHODOLOGY....................................275

9.1 INTRODUCTION 275

9.2 METHODOLOGY....................................275

9.2.1 Method: Qualitative 276
9.2.2 Research: Non-experimental 277
9.2.3 Approach: Phenomenological 278
9.2.4 Design: Case Studies....................................278

9.3 DETAILED RESEARCH DESIGN....................................286

9.4 SUMMARY AND CONCLUSION 289

9.4.1 Summary....................................289
9.4.2 Conclusion 290

10 CASE STUDY A....................................291

10.1 INTRODUCTION 291

10.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S) 291

10.3 FIRST CONTACT SESSION 292

10.4 ANALYSIS AND APPLICATION OF GATHERED DATA....................................299

10.5 SECOND CONTACT SESSION 302

10.6 MODIFICATIONS OF THE REAM....................................307

10.7 SUMMARY AND CONCLUSION 308

10.7.1 Summary....................................308
10.7.2 Conclusion 309

11 CASE STUDY B....................................310

11.1 INTRODUCTION 310

11.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S) 310

11.3 FIRST CONTACT SESSION 311

11.4 ANALYSIS AND APPLICATION OF GATHERED DATA....................................318

11.5 SECOND CONTACT SESSION 321

11.6 MODIFICATIONS OF THE REAM....................................325

11.7 SUMMARY AND CONCLUSION 326

11.7.1 Summary....................................326
11.7.2 Conclusion 327

12 CASE STUDY C....................................328

12.1 INTRODUCTION 328

12.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S) 328

12.3 FIRST CONTACT SESSION 329

12.4 ANALYSIS AND APPLICATION OF GATHERED DATA....................................335

12.5 SECOND CONTACT SESSION 338
12.6 MODIFICATIONS OF THE REAM 341
12.7 SUMMARY AND CONCLUSION 341
12.7.1 Summary 341
12.7.2 Conclusion 343
13 EVALUATION AND REVISED METAMODEL 344
13.1 INTRODUCTION 344
13.2 CASE STUDY INPUTS 344
13.3 EVALUATION OF INPUTS AND MODIFICATIONS OF THE REAM 345
13.3.1 Depiction Inputs 345
13.3.2 Clarification Inputs 350
13.3.3 Additional Feature Inputs 352
13.3.4 Expansion Possibility Inputs 353
13.4 MODIFIED REAM 355
13.5 SUMMARY AND CONCLUSION 359
13.5.1 Summary 359
13.5.2 Conclusion 359
14 THE REAM AT THE UNIVERSITY OF PRETORIA 360
14.1 INTRODUCTION 360
14.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S) 360
14.3 FIRST CONTACT SESSION 361
14.4 ANALYSIS AND APPLICATION OF GATHERED DATA 364
14.5 SECOND CONTACT SESSION 367
14.6 SUMMARY AND CONCLUSION 370
14.6.1 Summary 370
14.6.2 Conclusion 371
15 SUMMARY AND CONCLUSIONS 373
15.1 INTRODUCTION 373
15.2 RESEARCH FINDINGS 373
15.2.1 Detailed Findings 373
15.2.2 Broader Findings 403
15.3 EVALUATION OF THE RESEARCH 406
15.3.1 Methodology 406
15.3.2 Value and Contribution 407
15.4 RECOMMENDATIONS 408
15.5 CONCLUDING REMARKS 409
ADDENDUM A:8.1 GLOSSARY OF TERMS OF THE REAM 411
ADDENDUM A:9.1 SAMPLE INTRODUCTORY LETTER 416

ADDENDUM A:9.2 BACKGROUND SUMMARY 418
ADDENDUM A:9.3 FIRST SESSION INTERVIEW QUESTIONS............................421
ADDENDUM A:9.4 SAMPLE TEMPLATE FOR APPLIED REAM............................423
ADDENDUM A:9.5 SECOND SESSION INTRODUCTION.....................................424
ADDENDUM A:9.6 SECOND SESSION INTERVIEW QUESTIONS 434
ADDENDUM A:9.7 SAMPLE SIGN-OFF REQUEST LETTER.................................435
ADDENDUM A:10.1 SECOND SESSION – APPLICATION OF THE REAM 436
ADDENDUM A:11.1 SECOND SESSION – APPLICATION OF THE REAM 440
ADDENDUM A:12.1 SECOND SESSION – APPLICATION OF THE REAM 444
ADDENDUM A:14.1 BACKGROUND SUMMARY (UP)..448
ADDENDUM A:14.2 SECOND SESSION INTRODUCTION (UP) 451
ADDENDUM A:14.3 SECOND SESSION – APPLICATION OF THE REAM (UP).....457
REFERENCES..462

LIST OF TABLES

Table 1-1: Articles per Database 6
Table 2-1: Terms used in Enterprise Architecture Definitions 24
Table 2-2: Roles of Enterprise Architecture 75
Table 2-3: Benefits of Enterprise Architecture 76
Table 2-4: Benefits of Documenting Enterprise Architecture 77
Table 2-5: Challenges in Documenting Enterprise Architecture 77
Table 3-1: Analysis of Business Architecture Definitions 83
Table 3-2: Overview of Levels of Frameworks and Models (Business Architecture) 105
Table 4-1: Analysis of Information Architecture Definitions 112
Table 4-2: Overview of Levels of Frameworks and Models (Information Architecture) ... 134
Table 5-1: Analysis of Technology Architecture Definitions 140
Table 5-2: Overview of Levels of Frameworks and Models (Technology Architecture)... 159
Table 6-1: Proposed Taxonomy for Enterprise Architecture 169
Table 6-2: Domain Classification Guidelines (Bruls *et al.*, 2010: 528) 197
Table 6-3: Summary of Integration Strategies and Interlinking Mechanisms 212
Table 8-1: EA Artefact Template 248
Table 8-2: Description of Areas of the REAM 250
Table 8-3: Format for Principles 252
Table 8-4: Sample Principles for Process & Governance 253
Table 8-5: Sample Principles for Business Architecture 254
Table 8-6: Sample Principles for Information Architecture 255
Table 8-7: Sample Principles for Technology Architecture 256
Table 8-8: Sample Principles for Solution Architecture 257
Table 8-9: Examples of Application of Interlinking 265
Table 8-10: Summary of Areas in the REAM 271
Table 9-1: Qualitative Non-experimental Research Methods (adapted from Edmonds & Kennedy, 2013: 111) 276
Table 10-1: Data Analysis Case Study A 301
Table 11-1: Data Analysis Case Study B 320
Table 12-1: Data Analysis Case Study C 337
Table 13-1: Summary of Areas in the Modified REAM 357
Table 14-1: Data Analysis University of Pretoria 366
Table 15-1: Roles of Enterprise Architecture 375
Table 15-2: Benefits of Enterprise Architecture 376
Table 15-3: Benefits of Documenting Enterprise Architecture 377
Table 15-4: Challenges in Documenting Enterprise Architecture 377

Table 15-5: Overview of Levels of Frameworks and Models (Business Architecture)..... 379
Table 15-6: Overview of Levels of Frameworks and Models (Information Architecture) . 381
Table 15-7: Overview of Levels of Frameworks and Models (Technology Architecture). 383
Table 15-8: Proposed Taxonomy for Enterprise Architecture 384
Table 15-9: Summary of Integration Strategies and Interlinking Mechanisms 387
Table 15-10: Summary of Areas in the REAM 394
Table 15-11: Summary of EA *Status Quo* in Case Studies 396
Table 15-12: Summary of Application Possibilities in Case Studies................................ 396
Table 15-13: Summary of Response to the REAM by Case Studies 397
Table 15-14: Summary of Perceived Limitations of the REAM by Case Studies............. 397
Table 15-15: Summary of Suggested Enhancements.. 399
Table 15-16: Summary of Areas in the Modified REAM.. 401
Table A:9.4-1: Sample Template for the Applied REAM 423

LIST OF FIGURES

Figure 1-1: Enterprise Architectural Domains........ 2
Figure 1-2: Average Search Results by Topic........ 8
Figure 1-3: Article Searches on Integration between Architectural Domains 9
Figure 2-1: Synthesised Definition of Enterprise Architecture 25
Figure 2-2: History of Architecture Frameworks (Schekkerman, 2004a: 7)........ 27
Figure 2-3: Hype Cycle for Enterprise Architecture (Burton & Allega, 2010: 9)........ 31
Figure 2-4: Enterprise Architecture Scorecard Framework (Schelp & Stutz, 2007: 9)........ 47
Figure 2-5: Role of Frameworks (Shah & El Kourdi, 2007: 38) 51
Figure 2-6: The Zachman Enterprise Framework (Zachman, 2011) 53
Figure 2-7: TOGAF Architectures (Morar, 2008: 7)........ 55
Figure 2-8: TOGAF ADM Phases (The Open Group, 2009c: 54)........ 56
Figure 2-9: TOGAF Governance (Morar, 2008) 57
Figure 2-10: DoDAF (USA. Department of Defence, 2007b) 58
Figure 2-11: DoDAF links between Views (USA. Department of Defence, 2007a: 1-8). ... 59
Figure 2-12: FEAF (USA. FEA Program Management Office, 2004: 7) 61
Figure 2-13: FEAF Matrix (Other Architecture and Frameworks, 2006)........ 62
Figure 2-14: WSDOT Architectural Reference Model (Washington State. Department of Transport, 2001: 29)........ 65
Figure 2-15: Example of Process Model (Bedwell, 2006: 17) 68
Figure 2-16: Example of Relational Model (*Experts Exchange*, *s.a.*) 69
Figure 2-17: Example of a CLD (Vail, 2002: 10)........ 69
Figure 2-18: Example of Object-Orientated Model (Borysowich, 2007) 70
Figure 2-19: Gartner's Representation of Enterprise Architecture (Gartner, 2007: 17) 72
Figure 2-20: Sessions's Representation of Enterprise Architecture (Sessions, 2007: 2) .. 72
Figure 2-21: Jonkers's Representation of Enterprise Architecture (Jonkers *et al.*, 2006: 64)........ 73
Figure 2-22: Representation of Enterprise Architecture from www.e-cio.org (Wu, 2007: 2)........ 73
Figure 2-23: Synthesised Definition of Enterprise Architecture 74
Figure 3-1: Gartner's Dimensions of EBA (Blechar, 2009: 4)........ 81
Figure 3-2: EA containing Business Architecture (Kang, Lee & Kim, 2010: 3275)........ 82
Figure 3-3: Synthesised Definition of Business Architecture........ 84
Figure 3-4: FEAF's BRM (USA. FEA Program Management Office, 2004: 10)........ 90
Figure 3-5: Business Viewpoint (Gartner, 2008: 5) 91
Figure 3-6: Aspects of Business Architecture (McWhorter, 2008: 11)........ 92
Figure 3-7: Microsoft Motion – High Level Generic Module Map (Lloyd, 2006: 7)........ 93
Figure 3-8: The New Business Architecture for UC (University of California, 2000: 7)...... 95

Figure 3-9: Agile Business Modelling Framework (redrawn from Anon, 2009) 96
Figure 3-10: The Concepts of BMM (Rosen, 2008: 7).. 98
Figure 3-11: Capability Heat Map (Keller, 2009) 100
Figure 3-12: Business Model for Fact-based Ontologies (Kang, Lee & Kim, 2010: 3276)101
Figure 3-13: Example of Fact Types (Kang, Lee & Kim, 2010: 3278) 102
Figure 3-14: Example of DOLCE (Gangemi *et al.*, 2005: 28) 103
Figure 3-15: Synthesised Definition of Business Architecture.. 104
Figure 4-1: Synthesised Definition of Information Architecture 113
Figure 4-2: The structure of DRM (Executive Office of the President of the United States, 2007: 8).. 119
Figure 4-3: Gartner: Evolving the Information Architecture and Infrastructure (Blechar, 2007: 7).. 120
Figure 4-4: Gartner: SOA challenges (Blechar, 2007: 5)... 122
Figure 4-5: The Common Knowledge Enterprise Model (Neaga & Harding, 2005: 1099)124
Figure 4-6: Scope of Information Architecture (Higgins & Hebblethwaite, 2006: 4)......... 125
Figure 4-7: Queensland Government Information Architecture Abstract Model (Higgins & Hebblethwaite, 2006: 11) 126
Figure 4-8: The SIA integrating IS, IT and IM strategy (Pai & Lee, 2005: 154) 127
Figure 4-9: Evernden Eight (Evernden & Evernden, 2003: 29) 128
Figure 4-10: Synthesised Definition of Information Architecture..................................... 133
Figure 5-1: Synthesised Definition of Technology Architecture 141
Figure 5-2: TOGAF: Technology Architecture (The Open Group, 2009c: 137) 146
Figure 5-3: FEA's TRM (USA. FEA Program Management Office, 2004: 19) 147
Figure 5-4: Gartner's IT Architecture Guideline Framework (Rosser, 2002: 2) 148
Figure 5-5: Gartner TA Organising Components (Robertson, 2005: 19)......................... 150
Figure 5-6: IT Framework (Carbone, 2004: 47)... 152
Figure 5-7: Strategic Technology Architecture Roadmap (Radhakrishnan, 2006) 154
Figure 5-8: Victoria University of Wellington TA (Bates & Nelson, 2009: 1) 155
Figure 5-9: Synthesised Definition of Technology Architecture 158
Figure 6-1: The Holistic View of EA (Schekkerman, 2004b: 13) 162
Figure 6-2: Gartner Enterprise Architecture Framework (Guevara & Robertson, 2011: 2)165
Figure 6-3: Sessions's Representation of Enterprise Architecture (2007)....................... 165
Figure 6-4: Jonkers's Representation of Enterprise Architecture (Jonkers *et al.*, 2006).. 166
Figure 6-5: Wu's Representation of Enterprise Architecture from www.e-cio.org (Wu, 2007).. 166
Figure 6-6: Alonso's Representation of Enterprise Architecture (Alonso et al., 2010)..... 167
Figure 6-7: Minoli's Layered Model of Enterprise Architecture (2008: 17)....................... 168
Figure 6-8: The Proposed Integrated Diagram of EA 170
Figure 6-9: An Example of Integration of Unit EAs (Hasselbring, 2000: 35).................... 172
Figure 6-10: Relationship Scale of Terms 174

Figure 6-11: Frequency of Terms in the Literature 175
Figure 6-12: Synthesised Definition of Relating of Architectural Domains 176
Figure 6-13: Zachman DNA Example (O'Rourke, Fishman & Selkow, 2003: 615).......... 179
Figure 6-14: Zachman DNA Interaction (O'Rourke, Fishman & Selkow, 2003: 625)....... 180
Figure 6-15: Revised Zachman Framework (Graves, 2010) 181
Figure 6-16: TOGAF Content Metamodel (The Open Group, 2009a) 183
Figure 6-17: TOGAF Enterprise Continuum (The Open Group, 2009a).......................... 184
Figure 6-18: TOGAF's Integration of Architecture Artefacts (The Open Group, 2009c: 64) 185
Figure 6-19: Example of Concept Hierarchy by Gerber, Kotzé & Van der Merwe (2010: 60) 186
Figure 6-20: DoDAF v1.5 All view and Linkages (adapted from USA. Department of Defence, 2007a)... 187
Figure 6-21: DoDAF v2.0 Architecture Viewpoints (Okon, 2010: 9) 188
Figure 6-22: FEA Segment Model (Sessions, 2007) 190
Figure 6-23: The Structure and Methodology of FEA (adapted from USA.Chief Information Officer Council, 2001: 26)... 191
Figure 6-24: Gartner Enterprise Solution Architecture (Guevara & Robertson, 2011: 8). 193
Figure 6-25: Domain Architecture Partitioning (adapted from: Bruls *et al.*, 2010: 519) ... 195
Figure 6-26: Domain Taxonomy (Bruls *et al.*, 2010: 525)... 196
Figure 6-27: Service Layered View (Lankhorst *et al.*, 2005: 86) 198
Figure 6-28: BA and TA linking (adapted from:SOA Consortium EA2010. Working Group, 2010: 4) 199
Figure 6-29: Separation of Model, View, Visualisation and Viewpoints (Lankhorst *et al.*, 2005: 153) 200
Figure 6-30: Classification of EA Viewpoints (Lankhorst *et al.*, 2005: 163) 200
Figure 6-31: Conceptual View of an IT/Business Alignment Operationalization (adapted fromSaat *et al.*, 2010: 17).. 203
Figure 6-32: Enterprise Architecture Framework (adapted from Rohloff, 2008: 562)...... 205
Figure 6-33: The CEiSAR Enterprise Architecture Cube (adapted from CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 30) 206
Figure 7-1: An Iterative Approach to Development (Kruchten, 2000)............................. 216
Figure 7-2: Conceptual Model of Architectural Description (IEEE, 2000: 5).................... 220
Figure 7-3: Views and Viewpoints (Mykitshyn & Rouse, 2007: 149) 223
Figure 8-1: Intersection Examples... 230
Figure 8-2: Design Alternative A 231
Figure 8-3: Design Alternative B 234
Figure 8-4: Design Alternative C 237
Figure 8-5: Hexagonal Prism Net... 240
Figure 8-6: Proposed Relational EA Metamodel (REAM) – Architectural Domains......... 241

Figure 8-7: Proposed Relational EA Metamodel (REAM) with Current and Future States added 242
Figure 8-8: Proposed Relational EA Metamodel (REAM) with Views added 243
Figure 8-9: Proposed Relational EA Metamodel (REAM) with Relations added 244
Figure 8-10: Proposed REAM with Numbered Areas 246
Figure 8-11: The Complete Relation Enterprise Architecture Metamodel 259
Figure 8-12: Photographs of paper models of the REAM (Roets, 2012) 260
Figure 8-13: Summary of the phases of TOGAF's ADM (The Open Group, 2009c: 54) . 268
Figure 8-14: Proposed Relational EA Metamodel 270
Figure 9-1: Adapted Research Design mapping (redrawn from: Thomas, 2011: 77) 282
Figure 13-1: REAM with Changes to States 346
Figure 13-2: REAM with Changes to Numbering 347
Figure 13-3: REAM with Changes to Views 348
Figure 13-4: REAM with Changed Linkages 349
Figure 13-5: Legend Note 350
Figure 13-6: Changed Levels 351
Figure 13-7: The Composite Modified REAM 356
Figure 13-8: Rummler Business System (based on Rummler & Brache, 1995: 10) 358
Figure 13-9: REAM(s) in a Context Diagram (adapted from Rummler & Brache, 1995: 10) 358
Figure 15-1: Synthesised Definition of Enterprise Architecture 374
Figure 15-2: Synthesised Definition of Business Architecture 378
Figure 15-3: Synthesised Definition of Information Architecture 380
Figure 15-4: Synthesised Definition of Technology Architecture 382
Figure 15-5: Proposed Integrated Depiction of EA 385
Figure 15-6: Relationship Scale of Terms 386
Figure 15-7: Synthesised definition of Relating of Architectural Domains 386
Figure 15-8: Design Alternative A 390
Figure 15-9: Design Alternative B 391
Figure 15-10: Design Alternative C 392
Figure 15-11: Proposed Relational EA Metamodel 393
Figure 15-12: The REAM v1.0 400
Figure 15-13: REAM(s) in a Context Diagram (adapted from Rummler & Brache, 1995: 10) 402
Figure A:9.2-1: Definition of Enterprise Architecture 418
Figure A:9.2-2: Definition Relating the Architectural Domains 418
Figure A:9.2-3: Sample Ontology 419
Figure A:9.5-1: Respective Introduction Slides 425
Figure A:9.5-2: Slide 1 – Background 426
Figure A:9.5-3: Slide 2 – REAM 427

Figure A:9.5-4: Slide 3 – REAM 427
Figure A:9.5-5: Slide 4 – REAM 428
Figure A:9.5-6: Slide 5 – REAM 428
Figure A:9.5-7: Slide 6 – REAM 429
Figure A:9.5-8: Slide 7 – REAM 429
Figure A:9.5-9: Slide 8 – Applied REAM 430
Figure A:9.5-10: Slide 9 – Applied REAM 431
Figure A:9.5-11: Slide 10 – Applied REAM 431
Figure A:9.5-12: Slide 11 – Applied REAM 432
Figure A:9.5-13: Slide 12 – Applied REAM 432
Figure A:9.5-14: Slide 13 – Applied REAM 433
Figure A:10.1-1: Slide 8 – Case Study A Applied REAM: Processes & Governance 436
Figure A:10.1-2: Slide 9 – Case Study A Applied REAM: Domains 437
Figure A:10.1-3: Slide 10 – Case Study A Applied REAM: States 437
Figure A:10.1-4: Slide 11 – Case Study A Applied REAM: Views 438
Figure A:10.1-5: Slide 12 – Case Study A Applied REAM: Interlinking 438
Figure A:10.1-6: Slide 13 – Case Study A Applied REAM: Solution Architecture 439
Figure A:11.1-1: Slide 8 – Case Study B Applied REAM: Processes & Governance 440
Figure A:11.1-2: Slide 9 – Case Study B Applied REAM: Domains 441
Figure A:11.1-3: Slide 10 – Case Study B Applied REAM: States 441
Figure A:11.1-4: Slide 11 – Case Study B Applied REAM: Views 442
Figure A:11.1-5: Slide 12 – Case Study B Applied REAM: Interlinking 442
Figure A:11.1-6: Slide 13 – Case Study B Applied REAM: Solution Architecture 443
Figure A:12.1-1: Slide 8 – Case Study C Applied REAM Processes & Governance 444
Figure A:12.1-2: Slide 9 – Case Study C Applied REAM: Domains 445
Figure A:12.1-3: Slide 10 – Case Study C Applied REAM: States 445
Figure A:12.1-4: Slide 11 – Case Study C Applied REAM: Views 446
Figure A:12.1-5: Slide 12 – Case Study C Applied REAM: Interlinking 446
Figure A:12.1-6: Slide 13 – Case Study C Applied REAM: Solution Architecture 447
Figure A:14.1-1: Definition of Enterprise Architecture 448
Figure A:14.1-2: Definition Relating the Architectural Domains 448
Figure A:14.1-3: Sample Ontology 449
Figure A:14.2-1: UP: Introduction Slide 451
Figure A:14.2-2: UP Slide 1 – Background 452
Figure A:14.2-3: Slide 2 – The Modified REAM 453
Figure A:14.2-4: Slide 3 – The Modified REAM 453
Figure A:14.2-5: Slide 4 – The Modified REAM 454
Figure A:14.2-6: Slide 5 – The Modified REAM 454
Figure A:14.2-7: Slide 6 – The Modified REAM 455

Figure A:14.2-8: Slide 7 – The Modified REAM 455
Figure A:14.2-9: Slide 8 – The Modified REAM 456
Figure A:14.3-1: Slide 9 – UP Applied REAM: Processes & Governance 457
Figure A:14.3-2: Slide 10 – UP Applied REAM: Domains 458
Figure A:14.3-3: Slide 11 – UP Applied REAM: States 458
Figure A:14.3-4: Slide 12 – UP Applied REAM: Views 459
Figure A:14.3-5: Slide 13 – UP Applied REAM: Interlinking 459
Figure A:14.3-6: Slide 14 – UP Applied REAM: Solution Architecture 460
Figure A:14.3-7: Slide 15 – UP Applied REAM: Detailed Example 460
Figure A:14.3-8: Slide 16 – UP Applied REAM: Area T7 and Roadmaps 461

ABBREVIATIONS

3-D	Three-Dimensional
ABM	Activity-Based Methodology
ADM	Architecture Development Method
AM	Agile Modelling
ARB	Architecture Review Board
ARIS	Architecture of Integrated Information Systems
ASM	Architecture Specification Model
AV	All View
BA	Business Architecture
BEN	Business Engineering Navigator
BMM	Business Motivation Model
BPDM	Business Process Definition Metamodel
BPEL	Business Process Execution Language
BPMI	Business Process Management Initiative
BPMM	Business Process Maturity Model
BPML	Business Process Modelling Language
BPMN	Business Process Model and Notation
BRM	Business Reference Model
C4ISR	Command, Control, Communications, Computer, Intelligence, Surveillance and Reconnaissance
CADM	Core Architecture Data Model
CEiSAR	Centre d'Excellence en Architecture d'Entreprise
CEO	Chief Executive Officer
CIMOSA	Computer Integrated Manufacturing Open Systems Architecture
CIO	Chief Information Officer
CLD	Causal Loop Diagram
CMDB	Configuration Management Database
COBIT	Control Objectives for Information and Related Technology

COO	Chief Operating Officer
CORA	COmmon Reference Architecture
CRV	Common Requirements Vision
CTO	Chief Technology Officer
CWM	Common Warehouse MetaModel
DARS	Department of Defence Architecture Registry System
DAS	Direct Access Storage
DDPO	DOLCE + DnS Plan Ontology
DITA	Darwin Information Typing Architecture
DM	Data-Mining
DMZ	Demilitarised Zone
DNA	Deoxyribonucleic Acid
DNDCF	Department of National Defence and Canadian Forces
DnS	Descriptions and Situations
DoD	Department of Defence
DoDAF	Department of Defence Architecture Framework
DOLCE	Descriptive Ontology for Linguistic and Cognitive Engineering
DR	Disaster Recovery
DRM	Data Reference Model
DSL	Domain Specific Language
DTD	Document Type Definition
E2A	Extended Enterprise Architecture
E2AF	Extended Enterprise Architecture Framework
EA	Enterprise Architecture/Architect
EAA	Enterprise Application Architecture
EAI	Enterprise Application Integration
EAP	Enterprise Architecture Planning
EARB	Enterprise Architecture Review Board
EBA	Enterprise Business Architecture
EEM	Enterprise Engineering Methodology
EET	Enterprise Engineering Tools
EIA	Enterprise Information Architecture

EM	Enterprise Models
EML	Enterprise Modelling Languages
EOS	Enterprise Operational Systems
ERD	Entity Relationship Diagram
ESA	Enterprise Solution Architecture
ESAF	Enterprise Solution Architecture Framework
ESB	Enterprise Service Bus
ETA	Enterprise Technology Architecture
ExCo	Executive Committee
FaF	Facets are Fundamental
FEA	Federal Enterprise Architecture
FEAF	Federal Enterprise Architecture Framework
FOL	First-Order Logic
GEAF	Gartner Enterprise Architecture Framework
GEAM	Gartner Enterprise Architecture Method
GEMC	Generic Enterprise Modelling Concepts
GERA	Generic Enterprise Reference Architecture
GERAM	Generalised Enterprise Reference Framework and Architecture
GML	Generalised Markup Language
GRAAL	Guidelines Regarding Architecture ALignment
HL7	Health Level 7
HP	Hewlett-Packard
HR	Human Resources
HTML	HyperText Markup Language
IA	Information Architecture
IAF	Integrated Architecture Framework
IBM	International Business Machines Corporation
ICAM	Integrated Computer-Aided Manufacturing
ICT	Information and Communications Technologies
ID	Identity Document

IDEF	ICAM Definition
IEEE	Institute for Electrical and Electronics Engineers
IFEAD	Institute for Enterprise Architecture Developments
III-RM	Reference Model for Integrated Information Infrastructure
IM	Information Management
IP	Internet Protocol
IS	Information Systems
ISO	International Organisation of Standardisation
ISP	Internet Service Provider
IT	Information Technology
ITA	Information Technology Architecture
ITIL	Information Technology Infrastructure Library
J2EE	Java 2 Enterprise Edition
JTA	Joint Technical Architecture
KD	Knowledge Discovery
KPI	Key Performance Indicator
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
MDA	Model Driven Architecture
MDD	Model Driven Development
MHRS	Mariott Hotels, Resorts and Suites
MIS	Management Information System
MODAF	Ministry of Defence Architecture Framework
MOF	Meta-Object Facility
MSBA	Microsoft Services Business Architecture
MVC	Model View Controller
NAS	Network Attached Storage
NCR	National Cash Register Company
NCW	Net-Centric Warfare

OCCA	Open Cooperative Computing Architecture
OMG	Object Management Group
OWL	Ontology Web Language
PEM	Partial Enterprise Models
PERA	Purdue Enterprise Reference Architecture
PhD	Philisophiae Doctor
PKI	Public Key Infrastructure
POSIX	Portable Operating Interface for UNIX
PRM	Performance Reference Model
PSA	Project Start Architecture
RA	Reference Architecture
RACI	Responsible, Accountable, Consulted, Informed
RAG	Red, Amber, Green
REAM	Relational Enterprise Architecture Metamodel
RELAX NG	REgular LAnguage for XML Next Generation
RM-ODP	Reference Model for Open Distributed Processing
ROI	Return on Investment
SA	Solution Architecture
SABSA	Sherwood Applied Business Security Architecture
SAGA	Standards and Architectures of eGovernment Applications
SAN	Storage Area Network
SAP	Systems, Applications & Products in Data Processing
SBVR	Semantics of Business Vocabulary and Business Rules
SDLC	System Development Life Cycle
SIA	Strategic Information Architecture
SIB	Standard Information Base
SLiP	Sorta Like Python
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SOX	Schema for Object-Oriented XML

SPARC	Scalable Processor ARChitecture
SPP	Security and Privacy Profile
SQL	Structured Query Language
SRM	Service Component Reference Model
STAR	Strategic Technology Architecture Roadmap
SV	System & Services View
SVC	Solution Value Chain
SXML	Simple Extensible Markup Language
TA	Technology Architecture
TAFIM	Technical Architecture Framework for Information Management
TCP/IP	Transmission Control Protocol/Internet Protocol
TDWG	Taxonomic Databases Working Group
TEAF	Treasury Enterprise Architecture Forum
TISAF	Treasury Information Systems Architecture Framework
TOGAF	The Open Group Architecture Framework
TOM	TDWG Ontology Metamodel
TRM	Technical Reference Model
TV	Technical Standards View
UC	University of California
UML	Unified Modelling Language
UP	University of Pretoria
USA	United States of America
VLAN	Virtual Local Area Network
W3C	World Wide Web Consortium
WfMC	Workflow Management Coalition
WSCL	Web Services Communication Language
WSDL	Web Services Description Language
WSDOT	Washington State, Department of Transport
XMI	XML Metadata Interchange

XML	Extensible Markup Language
XSD	XML Schema Definition
Zachman DNA	Zachman Depth iNtegrating Architecture

1 INTRODUCTION

The context of the research will be to design an integrated metamodel for documenting and modelling of enterprise architecture (EA), encompassing the different architectural domains. The emphasis will be on the interlinking or interfaces between the different architectural domains. This lies within the fields of both Information Science and Information Technology (IT) and has incidence with Industrial Engineering. It can also be seen in the context of Enterprise Engineering – a "holistic approach to address enterprise changes, of all sizes and in all kinds of enterprises" (Dietz *et al.*, 2013: 92).

In this chapter, the research problem and questions will be discussed first, followed by the justification for and relevancy of this research. An initial literary review will be provided as well as a brief description of the different research methodologies. The assumptions to be used as a point of departure and the demarcation of the research area will also be listed. The outlay of the sections and individual chapters will be provided.

1.1 THE RESEARCH PROBLEM

The main research question is:

What should the key characteristics be of a metamodel for enterprise architecture, which focuses on the interfaces between the different architectural domains?

The main research question is subdivided into a set of sub-research questions: *a)* through *s)*. The sub-research questions will be described under the different headings of the three sections of the study.

The first section will entail a critical analysis of the existing literature, addressing the following sub-research questions:

*a) What definitions, frameworks and models are there for **enterprise architecture**?*
*b) What are the rationale, **purpose** and role of enterprise architecture?*
*c) What are the benefits and challenges in **documenting** enterprise architecture?*
*d) What definitions, benefits and models are there for **business architecture**?*

e) *What definitions, benefits and models are there for **information architecture**?*

f) *What definitions, benefits and models are there for documenting **technology architecture** or technical architecture?*

g) *What are the **taxonomy** and **relationships** of the different architectural domains?*

h) *What definitions, benefits and models are there for **integrating**, interacting and/or interlinking the architectural domains or are utilised for indicating the relationships between the architectural domains?*

Figure 1-1 is a preliminary possible representation of the architectural domains of enterprise architecture. The term 'solutions architecture' has been developed by Gartner (Guevara & Robertson, 2011: 1).

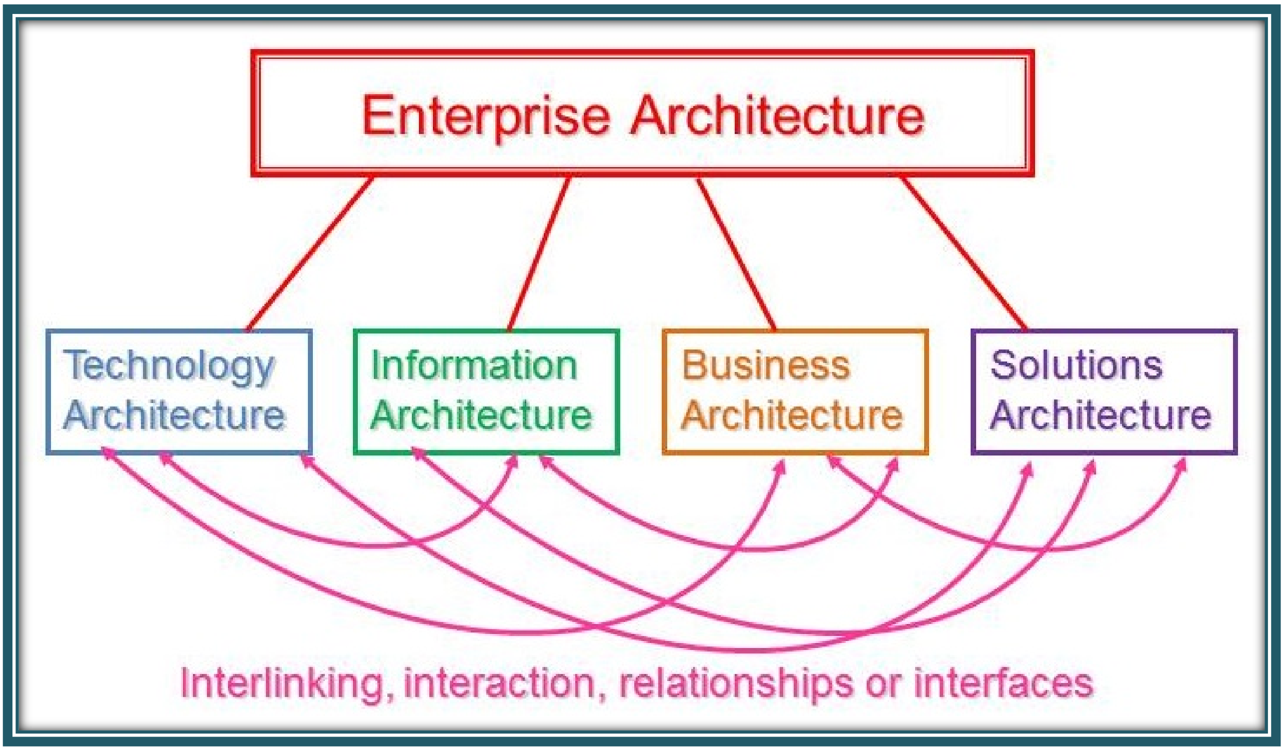


Figure 1-1: Enterprise Architectural Domains

The second section will involve the design of an integrated metamodel as a tool to describe holistically, plan and design the architecture of the enterprise. The integrated metamodel will use the literature analysis as a basis of departure. The purpose is to provide a simple but comprehensive tool to model the enterprise architecture, focusing on the links between the different architectural domains. It will include the modelling methodology, the modelling process, the modelling elements and deliverables

(Evernden & Evernden, 2003: 193), such as design principles, applicable standards, definition of terms, *etc.* The metamodel is then created according to the above-mentioned design methodology and process.

Sub-research questions to be answered as part of the second research component are:

i) *What will the **modelling** process look like, including the elements and deliverables which need to be addressed?*
j) *How will an integrated metamodel, interlinking the different architectural domains, **be constructed** and described?*

The third section will entail the description of the chosen methodology for the assessment of the designed conceptual metamodel by three different companies. The companies will be in different industries. This will test the practical application of the model through case studies. The assessment will lead to modifications and alterations to, and/or confirmation of the proposed metamodel. The modified metamodel will then be conceptually applied to the University of Pretoria. The University could find the metamodel beneficial and possibly utilise the outcome to enhance the understanding of its enterprise architecture and architecture practises in the different architectural domains.

Sub-research questions to be answered as part of the third research section are:

k) *What is the **status quo** of enterprise architecture in the case study enterprises?*
l) *What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*
m) *How was the proposed metamodel **received** in the case study enterprises?*
n) *What are the possible **limitations** of the proposed metamodel within the case study enterprises?*
o) *How was the proposed metamodel **modified**, based on the input from the case studies?*
p) *What is the applicability of the metamodel to the **University of Pretoria**?*
q) *What are the **conclusions** of the research?*
r) *What are the contributions of the research to the body of knowledge?*
s) *What future research possibilities flow from this research?*

1.2 JUSTIFICATION AND RELEVANCY

According to Kappelman (2010d: 27) it is not easy to choose between all the options, models and frameworks for enterprise architecture. He states that: “This is particularly true since we are still ‘early in the game’ and have little data by which to judge and evaluate different approaches.” It is thus evident that even in 2010, this field of research still needed research and attention.

Kappelman (2010d: 35) also reasons that all new advances take about three or four generations to become widely accepted and accepted as standard practices. As enterprise architecture is a new way of thinking and managing, there is still a great deal that needs to be learned and researched. Standards and generally accepted definitions do not exist either (2010d: 1).

Lankhorst *et al.* are of the opinion that companies are aware of and have recognised the need for an integrated architectural approach (2005: 47). The company's awareness leads to the development of their own architectural practice. There is a lack of a well-defined standard vocabulary or language, best practices and drawing standards in the design phase.

Lankhorst *et al*. also state that: “enterprise architectures often comprise many heterogeneous models and other descriptions, with ill-defined or completely lacking relations, inconsistencies, and a general lack of coherence and vision” (2005: 48). An integrated approach is necessary to determine the relations and their use.

According to a news article by Rasool, one of the big five issues on the agenda of CIO's (Chief Information Officers) for 2011 is “the role of enterprise architecture in driving innovation across Africa” (2011: 1). This indicates that enterprise architecture is still a burning issue and the application thereof is still to be figured out.

Baudoin describes the immaturity of enterprise architecture by means of an analogy (2010: 1). He puts the age of enterprise architecture at 23, based on the publishing of the first version of the Zachman framework in 1987. Although 23 is over the legal age for drinking at a bar, enterprise architecture still frequently has to prove and earn recognition – like showing an ID (Identity Document).

There is a noticeable gap in the literature regarding the integration of the architectural domains. Please refer to paragraph 1.3 for literature review.

The proposed study can contribute to this still maturing field of study, in providing an integration of different models and definitions and providing mechanisms to integrate the different architecture descriptions. This research is thus <u>relevant</u> and <u>justified</u> because

- it is still "early in the game";
- standards and definitions are not widely accepted;
- there is a lack of well-defined standard vocabulary;
- there is a lack of relationship models;
- the role of enterprise architecture is one of the big five issues for CIO's;
- enterprise architecture still needs to justify recognition; and
- there is a gap in literature regarding interlinking or interfaces between the different architectural domains.

1.3 LITERATURE REVIEW

A more detailed critical literature analysis will be provided in the thesis in Chapters 2 through 6. This chapter only contains a high-level overview on the literature available on the different subtopics, as detail will be provided in the following five chapters.

In order to obtain a general idea of the depth of available literature on the different sub-sections, a high-level survey was conducted on a variety of databases. The respective phrases, for example, 'Information Technology Architecture', were entered as search terms on each of the following databases: Infotrac, IngentaConnect, Ebscohost, Emerald, ISI Web of Knowledge, Scopus and SpringerLink. The phrases are grouped together into five bundles and colour-coded, *i.e.* technology (blue), information (green), business (orange), solutions (purple) and enterprise architecture (red). This colour convention, including pink indicating interlinking, will be applied to all tables, figures and models throughout the thesis. The results are tabled below in Table 1-1. Please note that, in the table, A represents the term Architecture.

Phrase	InfoTrac	Ingenta-Connect	EbscoHost	Emerald	ISI Web of Knowledge	Scopus	Springer-Link
Information Technology A	0	6	253	36	16	57	49
Technical A	114	29	280	101	60	258	561
IT A	80651	18	75911	8826	100000	236	563
Hardware A	175	115	1059	36	742	3642	2393
Software A	646	355	5844	220	2254	12036	8207
Storage A	252	15	523	4	73	372	291
Product A	46	71	365	141	136	384	512
Information A	299	103	2964	382	252	800	932
Systems A	573	82	3442	233	387	16721	2063
Data A	111	27	215	48	72	339	333
Application A	1089	26	505	59	97	609	0
Middleware A	17	10	140	12	130	528	699
Retrieval A	1	5	30	6	73	54	74
Information Management A	299	103	50	4	28	65	82
Business A	40	16	177	82	35	144	353
Solutions A	14	0	23	3	0	121	16
Enterprise A	517	54	1367	116	146	816	1109
TOTAL	**84844**	**1035**	**93148**	**10309**	**104501**	**37182**	**18237**

Table 1-1: Articles per Database

A summary of these results, based on the averages across the databases, is depicted in Figure 1-2 below. The purpose of the graph is to provide a visual overview of the distribution of academic articles on the relevant topics across the above-mentioned seven databases. It is evident that there are a large number of articles on IT architecture. The reasons for this could include the history of EA. According to Ross, Weill and Robertson: “Most of the effort to define enterprise architecture has been located in companies’ IT units” (2006: vii). Sessions poses that EA “was inaugurated to address two major problems in information technology (IT) … managing the increasing complexity of information-technology systems … the increasing difficulty in delivering real business value with those systems” (2007). This could result in a technology focus on EA activities. Despite the high numbers for IT architecture, the gap in articles regarding the overall interlinking of architectural domains remains (see also Figure 1-3). The searches were conducted during February 2011.

It is, however, interesting to note what similar searches on Google Scholar produces, though being unmoderated and including citations. The results were: Information Technology Architecture (496), Technical Architecture (4270), IT Architecture (3670), Hardware Architecture (15400), Software Architecture (39100), Storage Architecture (457), Product Architecture (3780), Information Architecture (8050), Systems Architecture (13700), Data Architecture (2270), Application Architecture (6670), Middleware Architecture (3800), Retrieval Architecture (287), Information Management Architecture (494), Business Architecture (2310), Solutions Architecture (184) and Enterprise Architecture (6920). Although the numbers are high, a distribution pattern similar to the academic databases is observed.

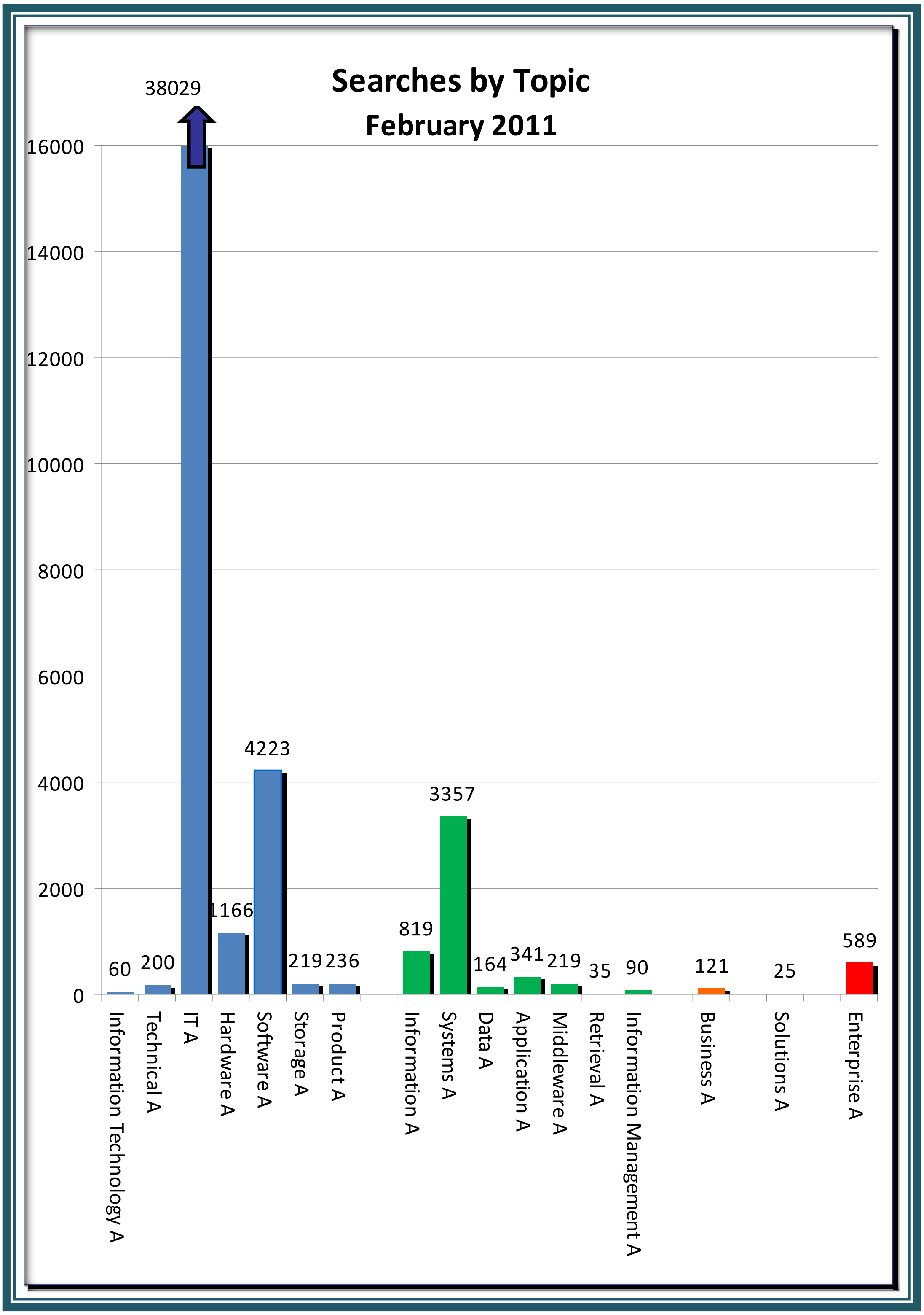


Figure 1-2: Average Search Results by Topic

The following deductions can be made from the search results depicted in the graph:

- The highest number of articles is available on Technology Architecture and especially on the phrases “IT Architecture” and “Software Architecture”.
- The second highest number of articles is available on Information Architecture.

- Very little literature is available on Solutions Architecture or Business Architecture.
- The term "IT Architecture" is much more widely used than "Information Technology Architecture".
- It is easier to get control over and describe architecture of a specific component, like hardware, software or systems than a widespread and varying architecture like business architecture.

The emphasis of this study is, however, on the interfaces or interlinking between the different specific architectures. Therefore another set of searches was performed in an attempt to find relevant literature. The same databases as mentioned above were consulted. The hits for each search were evaluated, by making use of the abstract, to determine possible relevancy. The results are portrayed in Figure 1-3 below. When these results are compared to Figure 1-2 it is evident that there exists a gap in the literature.

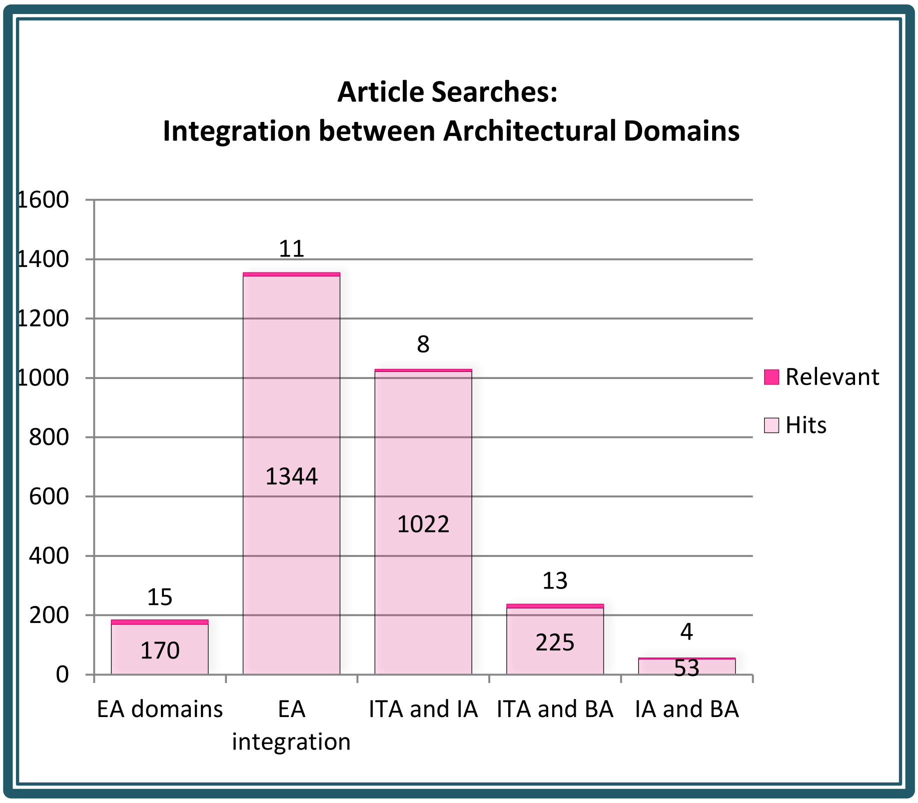


Figure 1-3: Article Searches on Integration between Architectural Domains

A status review of the most recent relevant literature is provided according to the different subsections of the research topic, including the important reference works as well as possible gaps.

1.3.1 Enterprise Architecture

Enterprise architecture is believed to have started with the Zachman framework (Baudoin, 2010: 1) and it is still widely used (Zachman, 2010: 39). This includes variations and applications of the Zachman Framework, for example the Ptech Zachman Framework Matrix (Vail, 2002: 9).

According to Schekkerman (2004b: 29) the Extended Enterprise Architecture (E2A) Framework includes all the elements for an overall holistic Enterprise Architecture. He also describes the TEAF (Treasury Enterprise Architecture Framework) as implemented in the United States (Schekkerman, 2004b: 113). The Integrated Architecture Framework (IAF) was developed by Capgemini (Schekkerman, 2004b: 139). Schekkerman discusses other examples like the Joint Technical Architecture (JTA) (2004b: 145) and the C4ISR (Command, Control, Communications, Computer, Intelligence, Surveillance and Reconnaissance) (2004b: 157) of the US Department of Defence.

Blevins, Spencer & Waskiewicz (2004: 2) refer to TOGAF's (The Open Group's Architecture Framework) ADM (Architecture Development Model) as a set of tools to develop an enterprise architecture and focus especially on the business requirements. It is useful for describing the enterprise architecture thus populating a framework with architectural models. Other frameworks are DoDAF (Department of Defence Architecture Framework) (USA. Department of Defence, 2007a: 1), FEAF (Federal Enterprise Architecture Framework) (USA. FEA Program Management Office, 2004: 2), SAGA (Standards and Architectures of eGovernment Applications) (Schekkerman, 2004b: 191), WSDOT (Washington State, Department of Transport) (Washington State. Department of Transport, 2001: 29) and GERAM (Generalised Enterprise Reference Framework and Architecture) (Saha, 2007: 2).

Other examples are the Enterprise Wheel (plotting the different activities in an enterprise) (Kappelman, 2010d: 5), the Wobble Model (indicating alignment problems) (Ballengee, 2010b: 48), the Actor-Network View (making use of actors and negotiations) (Sidorova & Kappelman, 2010: 77) and Generic Enterprise Modelling (providing a language to name every item and relationship) described by Simons, Kappelman & Zachman (2010: 133).

1.3.2 <u>Business Architecture</u>

Business architecture (BA) is less widely written about than technology and information architecture and has nothing directly to do with technology. The important aspects of business architecture is to acquire a plan or strategy, to provide a stable business view which will make it easier to keep up with changes in the technology domain and third it provides the link and language that the business community understands (Cook, 1996: 25).

There are a number of possible frameworks, for example the Gartner Business Architecture Framework (Gartner, 2008a: 4), the Microsoft Motion Business Architecture Method (Lloyd, 2006) (currently known as Microsoft Services Business Architecture (Sehmi, 2008)), the New Business Architecture of the University of California (University of California, 2000), Agile Business Process Modelling Framework (Anon., 2009) and the BMM (Business Motivation Model) (Rosen, 2008: 6).

There are a few examples for modelling business architecture, for example Causal Loop Diagram (CLD) as described by Vail (2002: 8); or Systems Thinking by Gharajedaghi (2006: 152) by making use of iterative idealised design; and the Value Chain Relationships (Gharajedaghi, 2006: 173). Gharajedaghi uses business architecture "to define the nature of relationships among the three dimensions of technology, product and market". Process architecture forms part of business architecture and is important in creating a world-class enterprise (Kim & Lee, 1996: 23). Other examples are Pi-Calculus Process Algebra (Parrow, 2001: 480), BPMN (Business Process Model and Notation (Object Management Group, 2008b) and the Capability Model (Keller, 2009).

Fact-based ontologies (Kang *et al.*, 2010: 3275) and DDPO (DOLCE + DnS Ontology) (Damjanovic, 2010: 497) are examples of ontologies for business architecture.

1.3.3 Information Architecture

Different existing models for information architecture (IA) have been described in the literature. Data architecture / modelling have been done with different versions of IDEF (ICAM (Integrated Computer-Aided Manufacturing) Definition) as well as ERD (Entity Relationship Diagram) (Cook, 1996: 79). Other examples are the Evernden Eight Factor model explained by Evernden & Evernden (2003: 28). They (Evernden & Evernden, 2003: 89) also utilises the Information Map.

Schekkerman describes The Technical Architecture Framework for Information Management (TAFIM) (2004b: 173) and the Computer Integrated Manufacturing Open System Architecture (CIMOSA) (2004b: 175) as examples. The Gartner Information Architecture Framework (Blechar, 2007: 7), Service Orientated Architectures (SOA) (Rehan & Akyuz, 2010: 2607), the 'Facets are Fundamental' Framework (FaF) (Crystal, 2007), the Common Knowledge Enterprise Model (Neaga & Harding, 2005: 1098), the Information Architecture Abstract Model (Higgins & Hebblethwaite, 2006) and the Strategic Information Architecture (SIA) (Pai & Lee, 2005: 153) are frameworks used to describe Information Architecture.

The IEEE (Institute of Electrical and Electronic Engineers) Standard 1471-2000 provides a solid theoretical base for system architecture (Lankhorst *et al.*, 2005: 22). Another model for systems architecture is RM-ODP (the Reference Model for Open Distributed Processing) and is described by Lankhorst et al. (2005: 29). The MHRS (Mariott Hotels, Resorts and Suites) developed a model for systems architecture (Gharajedaghi, 2006: 263). According to Blevins, Spencer & Waskiewicz (2004: 2) Model Driven Architecture (MDA) is used for systems development from OMG (Object Management Group).

1.3.4 Technology Architecture

Kim and Lee reason that an enterprise striving to become a world-class enterprise has to have a strong, responsive IT foundation and architecture in place – this includes IT systems architecture (1996: 22). Some models or frameworks are described in the literature. The GRAAL conceptual framework was developed to achieve a uniform manner of describing alignment in different enterprises and is used to describe IT Architecture (Lankhorst *et al.*, 2005: 224). Other examples of frameworks are the Gartner IT Architecture Guideline Framework (Rosser, 2002: 1), the IT Framework by Carbone (2004: 10), the Strategic Technology Architecture Roadmap (STAR) patented by Radhakrishnan (2006) and the Technology Architecture Framework from the Victoria University (Bates & Nelson, 2009: 1).

The USA (United States of America) Department of Commerce makes use of a Seven Step Process to document their IT Architecture (2004: 2). These steps are:

- Defining the vision, objectives and principles;
- Characterising the IT baseline;
- Creating a target architecture;
- Determining the gaps between current and target architectures;
- Development of a migration plan;
- Implementation of migration plan;
- Regular reviewing and updating.

Akhavan, Jafari and Ali-Ahmadi utilise four dimensions to describe the IT infrastructure , namely the extent of the intrafirm infrastructure, the interfirm infrastructure, infrastructure flexibility and the extent of data integration (2006: 519). Other examples are the Value Chain Framework as described by Ballengee (2010a: 149), and the Four Dimensions Standards Approach identified by Boh & Yellin (2007: 166).

1.3.5 Relationships between the Architectural Domains

It seems problematic to obtain literature addressing the inter-relationships and interlinking of the different architectures. This may be the result of the interdisciplinary nature of the topic. A few possible relevant articles have been retrieved, namely:

- Interface Descriptions for Enterprise Architecture (Garg, Kazman & Chen, 2006)
- Information System Integration (Hasselbring, 2000)
- An Integrated View on Business – and IT-architecture (Rohloff, 2008)
- Domain Architectures as an Instrument to Refine Enterprise Architecture (Bruls *et al.*, 2010)
- Exploring the Relationship between Information Technology, Infrastructure and Business Process Re-engineering (Bhatt, 2000).

Expansions of the Zachman framework, such as the Zachman DNA (Zachman Depth iNtegrating Architecture) (O'Rourke, Fishman & Selkow, 2003: 586) and Graves's Revised Zachman Framework (Graves, 2010), could be useful. Other possible frameworks are the Gartner Enterprise Architecture Framework (GEAF) (James *et al.*, 2005: 2), Domain Architecture by Bruls *et al.* (2010: 518), Service Orientation (Lankhorst *et al.*, 2005: 86), the use of viewpoints (Lankhorst *et al.*, 2005: 147) or metamodels (Saat *et al.*, 2010: 18), Rohloff's EA Framework (Rohloff, 2008: 562) and the CEiSAR cube (Centre d'Excellence en Architecture d'Enterprise) (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 26).

The literature provides part of the background and context necessary to enable the goal of designing and testing an architecture framework, which will address especially the integration or linkages between the different architectural domains. Background on the definition and role of enterprise architecture, the documentation thereof and an understanding of existing frameworks and models are essential in order to create a feasible new or modified framework.

1.4 RESEARCH METHODOLOGY

This qualitative research will make use of the following methodologies:

- Literature study – a critical literature analysis of recent relevant literature sources;
- Action research – the development or design of a framework/model through synthesis (The modelling methodology is described in Chapter 7.);
- Empirical case studies – assessing the applicability of the created metamodel with architecture practices in companies, making use of structured interviews, complemented by document analysis. Information gathered during the case studies will be recorded according to the questions in the structured interview. See Chapter 9 for the detailed research methodology. The input gathered from the case studies will propose adjustments or modifications to the developed metamodel. The adjusted metamodel will also be assessed for application at the University of Pretoria.

1.5 POINTS OF DEPARTURE

The research will be executed with the following points of departure as background:

- Enterprise architecture is the overarching architecture;
- Systems architecture, data architecture, application architecture, retrieval architecture, information management architecture and middleware architecture are all viewed as subsets of information architecture;
- Information technology architecture, software architecture, hardware architecture, technical architecture, product architecture and storage architecture are all viewed as subsets of technology architecture;
- There are different interpretations and connotations for the various terms and a reasonable definition will be selected and adhered to throughout the research.
- Enterprise Architecture is assumed as a given, although there are critics opposing the success of EA (Gaver, 2012; Hinchcliffe, 2009).

1.6 DEMARCATION OF RESEARCH

The following demarcation will be used during the research process:

- No hypotheses are being tested in this research.

- The methodologies and tools to implement the process of compiling an enterprise architecture are excluded from this study.
- Enterprise architecture maturity models and assessment are excluded from this study.
- Each architectural domain's existing models will not be described in depth, but a synoptic summary will be provided for each example.
- Solutions architecture will be touched on only briefly.
- The focus will be placed on the interaction or integration between the enterprise architectural domains.

1.7 OVERVIEW OF CHAPTERS

The thesis is divided into three sections, namely: 'Literature Study', 'Design of a Relational Metamodel' and 'Empirical Research – Case Studies'. The contents of the thesis will be structured as follows:

1. Introduction

A. Literature Study

2. Enterprise Architecture
3. Business Architecture
4. Information Architecture
5. Technology Architecture
6. Relationships between the Architectural Domains

B. Design of a Relational Metamodel

7. The Modelling Methodology
8. The Relational Metamodel

C. Empirical Research – Case Studies

9. Research Methodology
10. Case Study A
11. Case Study B
12. Case Study C
13. Evaluation and Revised Metamodel

14. The REAM at the University of Pretoria

15. Summary and Conclusion

This concludes the introduction, comprising the research problem and questions, the justification and relevancy, a literature review, the research methodology, points of departure, the demarcation of research and an overview of the chapters. Section A, covering the background and critical literature analysis, will start with the next Chapter – Chapter 2.

2 ENTERPRISE ARCHITECTURE

2.1 INTRODUCTION

The current status of enterprise architecture is described by means of a critical analysis of existing literature. This will provide essential background to create an understanding of the environment and will be used as input in creating a metamodel to facilitate the integration between the different architectural layers.

The **first** sub-research question to be answered in this chapter is:

a) *What definitions, frameworks and models are there for enterprise architecture?*

The answer to this question will be provided by analysing the existing definitions of enterprise architecture and providing a short overview of the history of enterprise architecture. This will be followed by an overview of the most important existing frameworks and models. General consensus confirms that enterprise architecture provides a blueprint of an enterprise's business, data, applications, and technology (Armour, Kaisler & Liu, 1999: 35; Jonkers *et al.*, 2006: 63; Kappelman *et al.*, 2010: 100; O'Rourke, Fishman & Selkow, 2003: 7; Salmans & Kappelman, 2010: 169).

The **second** sub-research question to be answered is:

b) *What are the rationale, purpose and role of enterprise architecture?*

The answer to this question will be provided by describing the role of enterprise architecture as well as the different functions and characteristics thereof. The answers will be provided from the available literature, without proving the validity thereof with actual case studies.

The **third** sub-research question to be answered is:

c) *What are the benefits and challenges in documenting enterprise architecture?*

The importance of the documentation process, the impact of not documenting as well as the benefits and challenges of the process are described from the literature.

2.2 DEFINITIONS OF ENTERPRISE ARCHITECTURE

The concept 'enterprise architecture' has two obvious components, of which the first is **'enterprise'**. The dictionary definitions of an enterprise are: "an organisation, especially a business" (Cambridge Dictionaries Online, 2011), "a business or a company", "an undertaking, especially one of some scope, complication, and risk, a business organization" (Webster's New World Dictionary, 2011). In other words an enterprise can indicate an agency, a whole corporation, a division of a corporation, a specific department or a group of enterprises (Abdallah & Galal-Edeen, 2006: 1). "Enterprises are purposeful entities of human endeavour, and they come in a wide range of forms and dimensions", for example companies, governmental agencies, healthcare institutions and supply chains (Dietz *et al.*, 2013: 109). The Open Group "defines 'enterprise' as any collection of organizations that has a common set of goals. For example, an enterprise could be a government agency, a whole corporation, a division of a corporation, a single department, or a chain of geographically distant organizations linked together by ownership" (The Open Group, 2009c: 5).

The second component is **'architecture'**. Architecture is known to indicate "the complex or carefully designed structure of something" (Oxford Dictionaries, 2011). Armour, Kaisler & Liu (1999: 37) describe an architecture simply as "the description of the set of components and the relationship between them." Blevins, Spencer & Waskiewicz (2004: 8) provide two definitions for architecture, namely: "a formal description of a system, or a detailed plan of the system at component level to guide its implementation" and "the structure of components, their interrelationships, and the principles and guidelines governing their design and evolution over time". Orr (2003: 1) sees 'architecture' as a practical synonym for 'high-level design'. Lin & Dyck (2010) refer to architecture as a "logical construction blueprint".

There are many and varied definitions for and interpretations of the concept **'enterprise architecture'**. This can obstruct and confuse the benefits of enterprise architecture – it is therefore essential to define enterprise architecture in a succinct way for common understanding within the enterprise.

2.2.1 Definitions in the Literature

There are various definitions of enterprise architecture in the literature. A few definitions are discussed below in chronological order, to show the development of the concept over time.

Armour, Kaisler & Liu (1999: 35) talk about the concept Enterprise Information Technology Architecture, and see this as "the blueprint for creating enterprise-wide information systems, and as such, describes a set of information system architectures." They also refer to it as a meta-architecture. This definition refers to enterprise information technology architecture, which evolved into the term enterprise architecture. It also lacks attention to the business component of enterprise architecture.

The definition used by Vail (2001: 1, 2002: 8) is: "An Enterprise Architecture is a set of aligned business and IT models of an enterprise, as well as the key aspects of the governing processes needed to keep these models applied and in a usable format". There seems to be more emphasis on the *status quo* than on plans for the future state.

Stevenson (2003) provides a well-worded comprehensive definition of enterprise architecture as, "a complete model of the enterprise; a master plan which acts as an integrating force between aspects of business planning such as goals, visions, strategies and governance principles; aspects of business operations such as business terms, organization structures, processes and data; aspects of automation such as application systems and databases; and the enabling technological infrastructure of the business such as computers, operating systems and networks."

Schekkerman (2004b: 22) cites the Meta Group's definition: "Enterprise Architecture is the holistic expression of an organisation's key business, information, application and technology strategies and their impact on business functions and processes. The approach looks at business processes, the structure of the organisation, and what type of technology is used to conduct these business processes." This definition addresses the main areas of business, information and technology as well as the impact on business processes. The strategies should be used to create a blueprint for the future.

Lankhorst *et al.* (2005: 3) provide the following definition: "Enterprise architecture: a coherent whole of principles, methods, and models that are used in the design and realisation of an enterprise's organisational structure, business processes, information systems, and infrastructure."

The next definition is provided by Bernard (2005: 31) – "The analysis and documentation of an enterprise in its current and future states from an integrated strategy, business, and technology perspective." This definition is very concise, but does not indicate the active process aspects that well.

Bedwell's (2006: 5) definition of architecture is worded, including the square brackets, to attempt to encapsulate all perspectives: "providing a framework [using their terminology], to help them understand the complexity of their business [as a whole]. So they can guide and manage their business [both now and into the future]." This definition has a client orientation and by focusing on the business side, fails to mention the information and information technology components.

Boh & Yellin (2007: 165) state that "EA standards specify the logical organization of corporate IT infrastructure, enterprise data and information, and applications that support core business processes." It is, however, important to regard enterprise architecture as an active process and not an exercise in documenting the implemented base.

A Gartner research project (Lapkin *et al.*, 2008: 2) resulted in the following comprehensive definition: "Enterprise architecture is the process of translating business vision and strategy into effective enterprise change by creating, communicating and improving the key requirements, principles and models that describe the enterprise's future state and enable its evolution. The scope of the enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them." This definition touches on all the important sections of enterprise architecture, but is a bit cumbersome for everyday use and internalisation within the enterprise.

Gartner (Lapkin *et al.*, 2008: 3) also emphasises that enterprise architecture should be defined in terms of "verbs" and not as a "noun" – to move the focus from the deliverables/artefacts to the process of deriving and applying artefacts within the enterprise. Enterprise architecture should not be an isolated academic exercise, but is actively involved in solving problems and providing realisable implementations. Enterprise architecture should form part of the planning process, management disciplines, executing disciplines as well as governance processes of the enterprise.

The Department of National Defence and Canadian Forces (DNDCF) (2009: 6) defines enterprise architecture as: "A collection of strategic information that defines a business, the information and technologies necessary to operate the business, and the transitional processes necessary for implementing new technologies in response to the changing needs of the business. It is represented through a set of integrated blueprint[s]." The definition touches all the perspectives, but defines enterprise architecture as the information and not as an active process. They (2009: 7) also distinguish between three classes of architecture, namely: reference architecture (doctrinal basis), as-is architecture (current state) and target architecture (desired future state).

The content of the above definitions was analysed based on the terminology used. The matrix in Table 2-1 provides a summary of terms, synonyms and verbs used. The terms were grouped together under output formats, actions, objects and scope. The group objects were subdivided into processes, strategy and domains. The terms most frequently used are: enterprise (enterprise wide), technology infrastructure and business plans/strategy. Second most frequent terms relate to the business domain and models. There are quite a few instances of similar terms or synonyms under the lower scoring terms. For example: coherent whole and holistic expression, and manage and govern.

		Authors	Armour, Kais & Liu	Vail	Stevenson	Meta Group	Lankhorst *et al.*	Bernard	Bedwell	Boh & Yellin	Gartner	DNDCF
Output formats		Master plan			•							
		Blueprint	•									•
		Set of architectures	•									
		Model		•	•		•				•	
		Coherent whole					•					
		Holistic Expression				•					•	
		Key requirements									•	•
		Principles					•				•	
		Methods					•					
		Frameworks							•			
Actions		Create	•								•	
		Apply		•								•
		Design					•					
		Realise					•					
		Manage							•			
		Govern									•	
		Communicate									•	
		Organise								•		
		Analyse						•				
		Document						•				
		Integrate			•							
		Change/Innovate									•	•
		Impact				•						
		Align		•								
Objects	Processes	Governance		•	•							
		Business			•	•	•					
		General								•	•	
		Transitional										•
	Strategy /Plan	Business		•	•	•					•	•
		Information				•						•
		Integrated						•				
		Application				•				•		
		Information Technology		•		•						
	Domains	Information								•	•	•
		Information Systems	•				•					
		Data			•					•		
		Applications			•							
		Technology infrastructure			•	•	•	•		•	•	•
		Organisation structure			•	•	•					
		Business			•			•	•			•
		Interrelationships									•	
Scope		Current		o		o	o	•	•	o		o
		Future						•	•		•	
		Enterprise wide	•	•	•		•	•	•	•	•	
		People									•	
	o	Implied in definition	•	Mentioned in definition								

Table 2-1: Terms used in Enterprise Architecture Definitions

This analysis of terminology, used in existing definitions, can assist in deriving a new composite definition.

2.2.2 Synthesised Definition

A definition should address what the concept is, what the scope of the concept includes and what the results are (Gartner, 2007: 9). The dictionary describes definition as "a statement that explains the meaning of a word or phrase" and "a description of the features and limits of something" (Cambridge Dictionaries Online, 2011).

Using the literature and the analysis of existing definitions as a base to form an opinion, a new definition is proposed. Architecture is defined as a process, based 1) on the Gartner definition (Lapkin *et al.*, 2008) and 2) on the fact that a major portion of one of the leading frameworks, TOGAF, is dedicated to the process, namely the ADM (Architecture Development Methodology). The suggested definition for enterprise architecture is represented in Figure 2-1. A comprehensive definition will contain all the respective terms. For certain purposes, situations or enterprises the emphasis will differ and a term from each block may be applicable, for example: enterprise architecture is the process of modelling the future-state blueprints of the information of an enterprise.

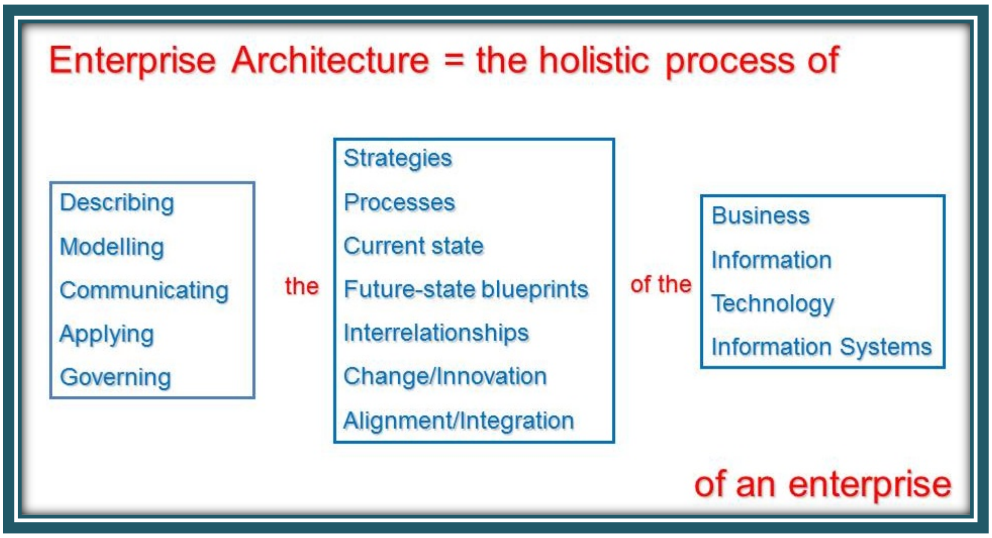


Figure 2-1: Synthesised Definition of Enterprise Architecture

The comprehensive written definition is:

> *Enterprise architecture is the process of describing, modelling, communicating, applying and governing the strategies, processes, current state, future-state blueprints, interrelationships, change/innovation and alignment/integration of the business, information, technology and information systems of an enterprise.*

The definition can be summarised as follows:

> *Enterprise architecture is the process of modelling and governing the current and future-state blueprints and interrelationships of the business, information and technology of an enterprise.*

Following the defining of the concept enterprise architecture is a short overview of its history and possible future.

2.3 HISTORY OF THE ENTERPRISE ARCHITECTURE CONCEPT

An overview of the history of enterprise architecture is described in terms of important frameworks or publications, marking growth points. The projected future of the different elements of enterprise architecture will also be highlighted.

2.3.1 History

The history of enterprise architecture is still relatively short and is linked to the development of frameworks. Enterprise architecture is widely deemed to have started with Zachman's framework, published in 1987 (Carbone, 2004: 11; Pessi, Magoulas & Hugoson, 2011: 54; Schekkerman, 2004b: 89). Schekkerman has done a study on the historic development of frameworks, which not only provides an overview of the history, but also the influence of frameworks on each other, supported by each other and adapted from each other. This is depicted in Figure 2-2.

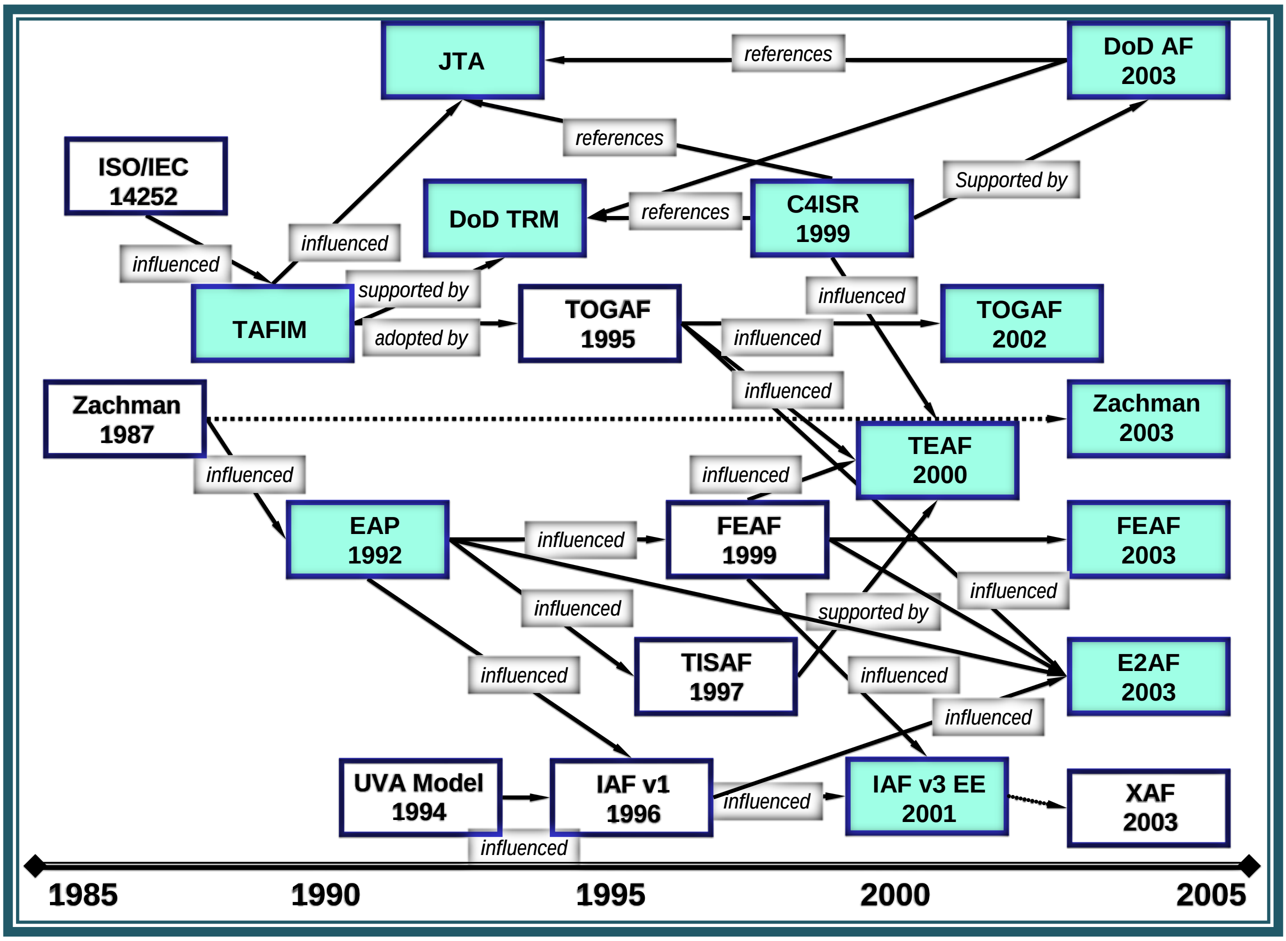


Figure 2-2: History of Architecture Frameworks (Schekkerman, 2004a: 7)

A rough timeline can be drawn up based on the above study and other sources:

- 1980: Porter published his book, *Competitive Strategy: Techniques for Analyzing Industries and Competitors* – this became a milestone work for strategic thinking, containing among others the idea of the value chain of an enterprise (Orr, 2003: 7).

First generation (focus on IT and business alignment): 1987 – 1994:

- 1987: John Zachman creates the Zachman Framework for Enterprise Architecture to establish a common vocabulary and set of perspectives for describing complex systems (Schekkerman, 2004b: 131). See 2.7.1 for detail.
- 1989-1992: Purdue Enterprise Reference Architecture (PERA) (Schekkerman, 2004b: 183).
- Late 80's: Technical Architecture Framework for Information Management (TAFIM) (Schekkerman, 2004b: 173). This framework was later retired and the work done was given over to TOGAF (The Open Group Architecture Framework) (Sessions, 2007).

- 1991: Computer Integrated Manufacturing Open System Architecture (CIMOSA) (Schekkerman, 2004b: 175).
- 1992: The methodology Enterprise Architecture Planning (EAP) was an attempt to provide guidance for the top two rows of the Zachman Framework and has a business data-driven approach (Schekkerman, 2004b: 101).

Second generation (creates a process to follow): 1995 – 2003:

- 1995: The Open Group Architecture Framework (TOGAF) was first developed based on TAFIM to provide an industry standard method of designing enterprise architecture (Schekkerman, 2004b: 119). TOGAF has continuously been updated and TOGAF version 9 is available today. See 2.7.2 for detail.
- 1995: The Command, Control, Communications, Computer Intelligence, Surveillance and Reconnaissance (C4ISR) framework was initiated and developed over the next few years. The focus was on interoperability and comparability across joint and combined enterprise boundaries (Schekkerman, 2004b: 157).
- Mid 90's – According to Westbrock (2007: 10) there existed five major frameworks, namely: Zachman, TOGAF, TAFIM, IBM's (International Business Machines Corporation) Open BluePrint and NCR's (National Cash Register Company) OCCA (Open Cooperative Computing Architecture).
- 1995: Benson and Parker's Square Wheel Framework, having a planning and an operations component in two domains – business and technology (Orr, 2003: 11).
- 1996: The US Department of Defence (DoD) released the Joint Technical Architecture (JTA) in August 1996. Several updated versions have appeared since (Schekkerman, 2004b: 145).
- 1999: The Federal Enterprise Architecture Framework (FEAF) is developed by the United States of America (USA) CIO (Chief Information Officer) Council as a result of the Clinger-Cohen Act passed in the United States of America. Development continued and in 2003 a newer version was published (Schekkerman, 2004b: 105). See 2.7.4 for detail.
- Late 90's: POSIX (Portable Operating Interface for UNIX) 1003.23 – architecture standard was developed by IEEE (Institute for Electrical and Electronics Engineers) and was administratively withdrawn in 2004. POSIX was aggressively business-

driven and existed in three phases, namely: a conceptual phase, a logical phase and lastly physical architecture (Lapkin, 2004: 3).

- 1999: Generalised Enterprise Reference Framework and Architecture (GERAM) was developed (Slay, 2002: 1). See 2.7.5.3 for a description.
- 2000: Department of Defence Technical Reference Model (DoD TRM) as predecessor of DoDAF (Department of Defence Architecture Framework) (Schekkerman, 2004b: 165).
- 2000: Standards and Architectures of eGovernment Applications (SAGA) of the German Federal Government (Schekkerman, 2004b: 191). See 2.7.5.2 for a description.
- 2002: Federal Enterprise Architecture (FEA) was commissioned by the government of the USA (USA. FEA Program Management Office, 2004: 2).
- 2002: Treasury Enterprise Architecture Framework (TEAF) was derived from TISAF (Treasury Information System Architecture Framework) and FEAF and was another USA Government effort to standardise enterprise architecture (Schekkerman, 2004b: 113).
- 2003: The Extended Enterprise Architecture (E2A) Framework was created by the Institute for Enterprise Architecture, based on other frameworks and practical experience. The E2A Framework is "dealing with the processes and activities of extending the Enterprise Architecture beyond its original boundaries, defining a collaborative environment for all entities involved in a collaborative process" (Schekkerman, 2004b: 93). See 2.7.5.1 for a description.
- 2003: DoDAF came into existence as an evolution of the C4ISR framework. "The framework provides rules and guidance for developing and presenting architecture descriptions" (Schekkerman, 2004b: 158). DoDAF v 1.5 was released in 2007 (USA. Department of Defence, 2007a: 3). See paragraph 2.7.3 for detail.
- In 2003, the Cutter Consortium (Orr, 2003: 1) reported that after a decade of limited attention or progress, the interest in modelling enterprises had escalated again. This might be due to new Internet technologies, an aging IT workforce with critical knowledge of legacy systems as well as better management of IT assets.

Third generation (faster results): 2005 - 2011:

- 2005: Gartner bought over the Meta Group and their enterprise architecture practice (Sessions, 2007) which was expanded to the Gartner Enterprise Methodology.

- 2008: There are 20 widely known frameworks, for example, TOGAF, TAFIM, C4ISR, JTA, DoDAF and Zachman Framework 2 (Westbrock, 2007: 10).
- 2008: The Zachman Framework[2] (Westbrock, 2007: 10).
- 2010: Rosen (2010: 1) reports on a survey done among 148 respondents: 34% are using TOGAF, 29% use a combination of frameworks, 16% do not use a framework, 3% use Zachman, 2% DoDAF and the rest other frameworks.

2.3.2 Future

What does the future of enterprise architecture look like? Newman *et al.*(2009: 1) predict major changes: "Accelerating change will have a major impact on traditional enterprise architecture (EA), as stakeholder interests become more divisive, interdependencies more complex and behaviours more chaotic."

Gartner (Burton & Allega, 2010) published a piece of research indicating the expected growth and decline in components of a technology or concept, being called a hype cycle. In Figure 2-3 their hype cycle for enterprise architecture is depicted. The hype cycle consists of a curve, indicating certain phases; markers, indicating the expected time frame and labels, indicating the different components.

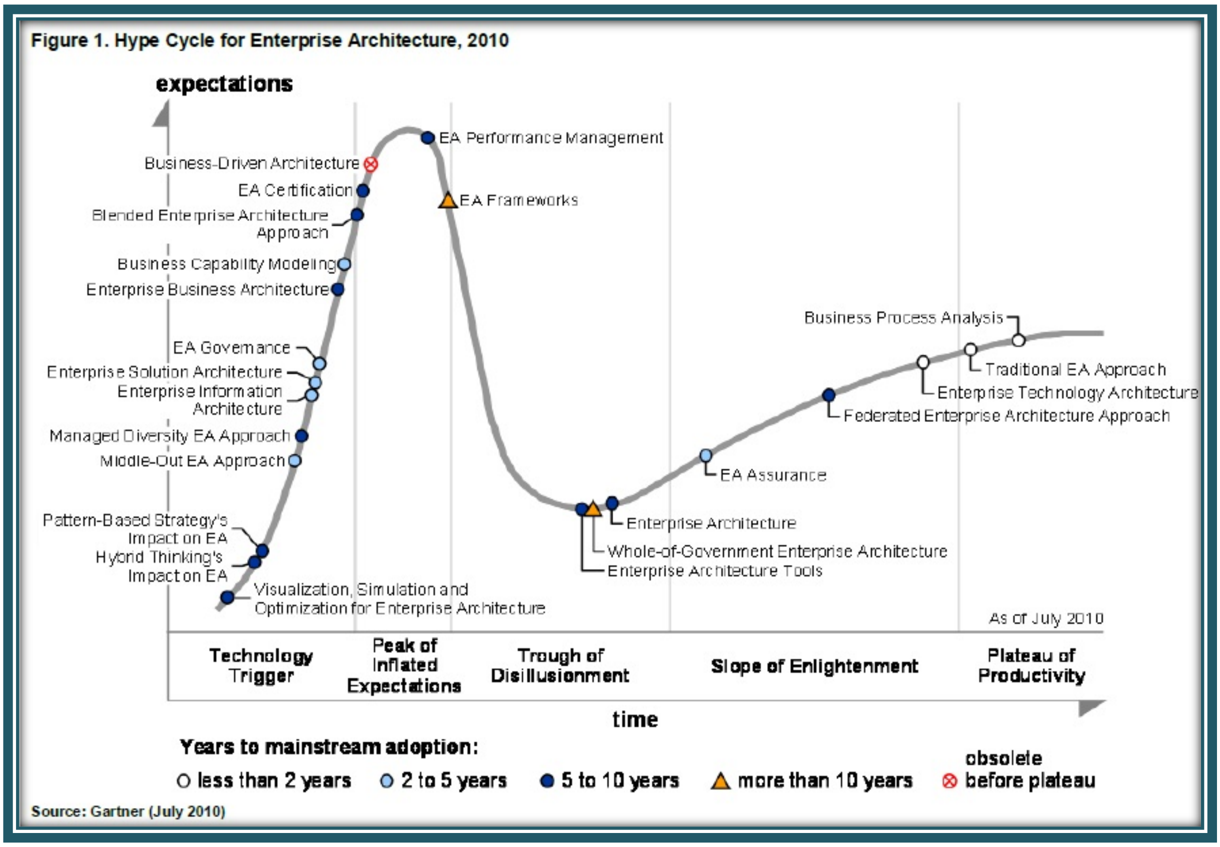


Figure 2-3: Hype Cycle for Enterprise Architecture (Burton & Allega, 2010: 9)

Some interesting observations from the hype cycle are:

- Business-Driven Architecture is suspected to become obsolete before reaching a plateau.
- EA frameworks and Whole-of-Government EA will probably need more than ten years to mature.
- The nearly mature components are Business Process Analysis, Traditional EA Approach and Enterprise Technology Architecture.
- Enterprise Architecture and its tools are in the trough of disillusionment due to the inability to really become integrated with the business (Burton & Allega, 2010: 4).
- There is, however, a cluster of entries on the trigger and trigger up slope, which illustrates a new attempt to integrate and engage with the business (Burton & Allega, 2010: 4).
- Eleven of the 23 components are suspected to take another five to ten years to reach a mature plateau.

Although enterprise architecture already started in 1987, it still has quite a few immature and new components to develop. It is also important to understand the role of enterprise architecture within an enterprise.

2.4 THE ROLE OF ENTERPRISE ARCHITECTURE

It is important for an enterprise to understand the role, functions and benefits of enterprise architecture, before investing and embarking on an enterprise architecture project or process. In other words: why enterprise architecture, what is the value proposition? Understanding and communicating these aspects will make the value of enterprise architecture within the enterprise more visible and practical. This section aims to describe the *raison d'aitre* of understanding the architecture of an enterprise.

In 2007 Salmans & Kappelman conducted a survey among IT enterprises to determine the state of enterprise architecture in these enterprises. They analysed 377 quality responses. Their analyses indicated that the respondents, in general, believed in the potentially positive role of enterprise architecture, for example, to facilitate change, to align initiatives and objectives, to provide a blueprint, to be used as a planning tool and a tool for decision-making (Salmans & Kappelman, 2010: 169). This potentially positive role has a few facets. "The most important deliverable of enterprise architecture is change. Without a real impact on the way the enterprise effectively changes, all those deliverables are for naught" (Lapkin *et al.*, 2008: 1).

One facet of the role of enterprise architecture is to provide a holistic **vision** for the enterprise. According to Schekkerman (2004b: 13) "a rigorously defined framework is necessary to be able to capture a vision of the '*entire organisation'* in all its dimensions and complexity." Enterprise architecture is able to provide this framework to capture the essence of an enterprise, by integrating all the facets in a holistic way. Cannon (2010: 122) puts it like this: "CIOs communicate the vision of a new 'system' from the computer room to the boardroom. In doing so, the CIO is using enterprise architecture to drive organization alignment". Enterprise architecture can be used for a holistic approach in contrast with different management nostrums being applied (Veasey, 2001: 420). Shah & El Kourdi (2007: 37) concur that enterprise architecture facilitates enterprise planning. The bulk of the effort in the enterprise architecture process should thus go into describing the future state or vision of the enterprise and not into describing the *status quo*.

Second, architecture comes into play when the object or enterprise is so **complex** that all the elements and detail cannot be comprehended or remembered without problems and oversights, as well as when change needs to be implemented to the object or enterprise (Zachman, 2010: 38). DeBoever, Paras & Westbrock found that meaningful enterprise architecture always focuses on reducing (not eliminating) the complexity of processes across the breadth of the enterprise (2010: 157). Veasey (2001: 420) is also of the opinion that enterprise architecture will provide tools for managing complexity. Reduction of complexity has many results and benefits, as described below in paragraph 2.5.

Another role of enterprise architecture is to create a **framework** and bundle of tools to facilitate practical implementation and application of the architecture blueprint. “Achieving this requires a collection of tools, practices, guidance, heuristics, and specialized content (patterns, frameworks, domain models *etc.*) assembled to support the enterprise organization as it designs, develops, deploys, manages, and evolves its solutions” (Brown, 2008: 178).

The threefold role of enterprise architecture is thus:

- to capture a holistic view and vision of the future state of the enterprise,
- to capture the complexity of the enterprise in a manageable fashion, and
- to create a framework and toolset for implementation and application.

2.5 THE FUNCTIONS AND QUALITIES OF ENTERPRISE ARCHITECTURE

There are a number of different functions, features or characteristics of enterprise architecture. The main functions are:

- Aligning between domains,
- Provisioning of information,
- Enabling of innovation,
- Improving agility of the enterprise,

- Improving the utilisation of resources,
- Enhancing of processes,
- Reducing of risks,
- Improving linking with external partners.

Each function will be discussed briefly, based on existing literature, in the following sub-sections. The approach to enterprise architecture, followed within an enterprise, will determine greater or lesser emphasis on each function.

2.5.1 <u>Aligning Domains</u>

Enabling of alignment at several levels is the penultimate purpose of enterprise architecture (Ballengee, 2010b: 46; Salmans, 2010: 89). Enterprise architecture can be utilised as an alignment tool (Kappelman *et al.*, 2010: 100), or a tool to design alignment architecture (Aier & Winter, 2009: 161). Enterprise architecture enables the ability for the enterprise to understand and determine the needs for alignment and integration (O'Rourke, Fishman & Selkow, 2003: 7). Enterprise architecture provides processes to develop an IT strategy and to assist aligning with business strategies and business implementations (Rico, 2006: i; Shah & El Kourdi, 2007: 37).

Vail (2002: 3) is of the opinion that enterprise architecture assists the enterprise in developing effective and integrated business and IT strategies as well as the sharing of knowledge, alignment and consistency between strategies. The pitfall of different business-units' IT functions operating at odds with one another can be avoided by following the enterprise architecture process (Schekkerman, 2004b: 25).

Alignment needs two equal counter parts, for example, IT and Business. Understanding of both areas is a prerequisite for establishing alignment between them – it is problematic to align with something vague and non-descript. The value of IT in business decisions needs to be recognised and *vice versa* (Baudoin, 2010: 8). Ross, Weill & Robertson (2006: 98-99) describe this alignment function in terms of the satisfaction of management. This entails the business' confidence in the IT unit's delivering ability and enterprise wide commitment to architectural improvements and changes. Part of the alignment benefit is having a shared terminology or language used in describing every aspect of an enterprise.

2.5.2 Provisioning Information

Enterprise architecture "yields centralized, stable, and consistent information about the enterprise environment" (Schekkerman, 2004b: 14). Examples are the identification of new markets, measurements of compliance to customer needs and identification of gaps in systems or services. Enterprise architecture enables different views of the different aspects and therefore provides various perspectives on the same information (Shah & El Kourdi, 2007: 37).

The architecture process can lead to "knowledge of the business, its value chain, and how value is created" (Ballengee, 2010b: 50). The IT side of the enterprise increases in business acumen and understanding of business jargon, while the business side can benefit from IT's perspectives and fresh approach. Kappelman *et al.* (2010: 101) agree when listing "better situational awareness" as well as "aligning business objectives with IT" as major benefits of enterprise architecture.

Information silos within an enterprise can hinder corporative initiatives, but by unlocking the power of information these silos can be unified. According to Schekkerman (2004b: 25), enterprise architecture should unlock this power of information. Kappelman (2010a: 35) is of the opinion that enterprise architecture can play an important role in bridging the gap or chasm between strategy and implementation. This could be achieved by capturing the knowledge of the enterprise and making it available in real time for every management need.

There are a number of specific applications of the availability of information within the enterprise, which will be highlighted below, *i.e.* using the information for decision support, as a planning tool and to enhance the satisfaction of customer requirements.

2.5.2.1 Decision Support

Enterprise architecture provides precise and high-quality information, which enables enterprises to make better and more agile decisions. This increases the ability to respond to the forces of change (Schekkerman, 2004b: 14). According to Brown (2008: 177) a standardised architecture provides assistance to decision-makers, which they can utilise in comparing the architecture of alternative systems and their design. For this purpose the architecture must be documented well enough to enable detailed analysis, which can be used in justifying procurement decisions and support.

Sidorova & Kappelman (2010: 70) states that the role of enterprise architecture in guiding management and technology decisions has long been established and provides several examples. This was also substantiated by a study by Kappelman *et al.* (2010: 100), which showed that enterprise architecture as a tool for decision-making was widely accepted. Investment decisions can also be maximised by using enterprise architecture (Cannon, 2010: 122).

2.5.2.2 Planning Tool

Zachman (1996: 6) explains that enterprise architecture can prevent making choices in a vacuum and position the enterprise to see a total range of alternatives. Kappelman *et al.* (2010: 100) ranked the function of enterprise architecture as a planning tool as the second highest function of enterprise architecture. Enterprise architecture can provide a unified perspective on the information for planning, including enterprise resource planning, manufacturing resource planning and choosing planning and scheduling software (Neaga & Harding, 2005: 1095).

2.5.2.3 Satisfaction of Customer Requirements

The enterprise architecture process will enable an enterprise to be in a better position to satisfy and comply with the customers' expectations and stakeholders requirements (Schekkerman, 2004b: 20). It can also provide a comprehensive view of enterprise wide requirements (Kappelman *et al.*, 2010: 97) and consolidate customer data across different business units and/or databases (Thundatil, 2007: 469).

More customer intimacy, including customer service, responsiveness and relationships, can be achieved based on a deeper customer knowledge (Ross, Weill & Robertson, 2006: 100). IT can use enterprise architecture to "better serve enterprise needs and to communicate with customers and stakeholders" (Kappelman, 2010b: 120).

2.5.3 Driving Innovation/Change

One of the functions of enterprise architecture is to provide a better understanding of change and its impact, as well as to drive innovation. Zachman (2010: 38) reckons that architecture is imperative to provide the ability to manage change and implement change in a minimum time, with minimum disruption at minimum cost. "Managers can thus choose to plan evolution, or they can react when reality hits and 'evolve' parts of the information systems according to the latest crisis" (Armour, Kaisler & Liu, 1999: 35).

Kappelman *et al.* (2010: 100) show that facilitating systematic change is one of the widely accepted functions of enterprise architecture. A well-defined architecture is an asset for positioning new developments, identifying necessary changes and facilitates innovation "by providing both stability and flexibility" (Jonkers *et al.*, 2006: 64).

DeBoever, Paras & Westbrock (2010: 160) describe the potential impact of enterprise architecture as providing a "process to analyze and plan for change intelligently and consistently across the entire enterprise with full knowledge of the impact of change as well as the consequences of not changing." There is a need for an integrated approach regarding change. This integration implies the establishment of coherence and consistency within the various business and organisational aspects, as identified in the architecture frameworks (Hoogervost, 2004: 228).

Enterprise architecture thus "develop(s) a proactive enterprise capable of meeting customer demands, outpacing the competition, and driving innovation" (Schekkerman, 2004b: 25). Lankhorst *et al.* (2005: 44) coined this feature as "innovation power", while Ross, Weill and Robertson (2006: 100) see this function as an opportunity to be first to market with innovative products and services.

2.5.4 Creating Agility

One of the inhibitors of change is complexity. Applying enterprise architecture can lower the complexity and thus result in making an enterprise more agile to adjust and to make decisions more quickly (Schekkerman, 2004b: 25). Baudoin (2010: 16) from the Cutter Consortium describes agility as “to enable new business capabilities through IT in a way that is rapid, safe, secure, economical, and does not make the next effort more complicated”.

Higher agility than competitors can help enterprises to enter markets earlier than their competitors (DeBoever, Paras & Westbrock, 2010: 161), (Ross, Weill & Robertson, 2006: 100). O’Rourke, Fishman & Selkow (2003: 7) use the term “responsiveness of the business to technology and to the marketplace” to describe this agility. Adding to this function of organisation architecture is to specifically increase IT responsiveness. IT responsiveness is enhanced by the fact that IT and business leaders have fewer technology choices due to a standardised environment “and thus spend less time making technology decisions or addressing unexpected technical problems” (Ross, Weill & Robertson, 2006: 96). The time to make decisions and implement them, which equals the responsiveness of IT, is thus shortened.

Brown (2008: 183) reasons that an integrated enterprise architecture will enable IT to respond quickly and appropriately to business requirements. This includes the improvement of system interoperability and flexibility (Kappelman *et al.*, 2010: 97, 100). Lankhorst *et al.* (2005: 44) also view interoperability and flexibility as a relevant benefit of enterprise architecture: “Flexibility to replace or substitute services in case of failure, flexibility to upgrade or change services without affecting the organisation’s operations, flexibility to change suppliers of services, flexibility to reuse existing services for the provisioning of new products or services”.

Chanopas, Krairit & Khang (2006: 638) describe a flexible IT architecture in terms of connectivity, compatibility, modularity, IT personnel competency, scalability, continuity, rapidity, facility (ease of use) and future survivability. Attention to all of these areas, throughout the enterprise architecture process, will ensure a more agile or flexible IT enterprise.

An outcome or consequence of this acquired agility is a reduction in delivery time. Enterprise architecture should "reduce solution delivery time and development costs by maximizing reuse of technology, information, and business applications" (Schekkerman, 2004b: 26). This function or benefit will be dependent on the maturity of enterprise architecture within the enterprise.

2.5.5 Improving Utilisation of Resources

Doing less with more or improving the utilisation of resources has been part of the management challenge for a long time. Shah & El Kourdi (2007: 37) view "resource management, task allocation and scheduling, and cost estimations" as major aspects of competitive advantage.

Lankhorst *et al.* (2005: 44) see "cost effectiveness" as one of the relevant benefits of enterprise architecture and Ross, Weill and Robertson (2006: 100) describe this as "*Better operational excellence:* low-cost, reliable, and predictable operations, with emphasis on cost".

The improved utilisation of resources is especially evident in the following areas, and will be described as such:

- utilisation of IT resources,
- reduction of IT costs, and
- reduction of duplication.

2.5.5.1 Utilising IT Resources

The effective and efficient use of technology within an enterprise is a key motive for setting up an architecture (Klouwenberg, Koo & Van Schaik, 1995: 9). The results of a study done by Kappelman *et al.* (2010: 101) show that "improved utilization of IT" as well as "more effective use of IT resources" are two of the five main benefits of enterprise architecture. The other three are: improved interoperability, aligning with business objectives and better situational awareness.

Boh & Yellin (2007: 187) did empirical research and proved that the use of enterprise architecture standards does indeed improve the sharing and integration of information technology resources. The most beneficial impact can be obtained in “managing physical and human IT infrastructure, followed by application integration, and finally enterprise data integration.” They (Boh & Yellin, 2007: 175, 187) also proved their hypothesis that enterprise architecture can identify and consolidate different systems used for the same function.

2.5.5.2 Reducing IT Costs

Cost is one of the major resources to be managed by an enterprise. Concerns or pressures to reduce IT costs is frequently a driver for enterprise architecture initiatives. Enterprise architecture introduces discipline in systems as well as processes, which results in controlling the high costs of business silos. This could result in the reduction of:

- IT operations unit cost – decreasing with architecture maturity; and
- application maintenance cost – time and total cost to apply changes.

For example: by moving to consolidated data centres, reducing the number of technologies in use and introducing standards, companies can on average lower their IT budget by 15% (Ross, Weill & Robertson, 2006: 93).

Rico developed a model to measure the ROI (return on investment) of enterprise architecture (Rico, 2006: iii). He used the “metrics: (a) costs; (b) benefits; (c) benefit to cost ratio; (d) return on investment; (e) net present value; and (f) breakeven point” and applied them to different types of costs, for example, reduced redundancy cost and economic development cost (Rico, 2006: v). The application of such a model requires a level of maturity and the availability of all the relevant figures.

Peslak (2008: 55) conducted a survey among top financial executives. He found that “the mere existence of a written strategic plan was sufficient to improve overall technology returns”, which improves still more with higher alignment. Lyn & Dick (2010) deduces from this that “if the enterprise architecture is developed with the organization’s IS strategic plan, aligned with its business plan, then that can translate into improved chances for an organization’s success through improved return on investment.”

2.5.5.3 Reducing Duplication

During the enterprise architecture process, enterprises become aware of duplications and inconsistencies. In addressing these issues the technology can be streamlined to reduce complexity and cost (Schekkerman, 2004b: 14). This will also result in a decrease in the yearly support cost.

Enterprise architecture can provide time savings to all business areas by preventing decisions and implementations that are redundant or too narrowly focused (DeBoever, Paras & Westbrock, 2010: 160). By providing a bigger picture and awareness, enterprise architecture can reduce duplication by reducing complexity and providing a broader focus.

2.5.6 Enhancing Processes

A significant function of enterprise architecture is enhancing different processes within the enterprise, *i.e.* communication, architecture, strategic governance and business processes.

2.5.6.1 Communication Processes

Enterprise architecture provides a mechanism to be used for enabling communication. This communication will focus on the important elements of the enterprise as well as the functioning of the enterprise (Schekkerman, 2004b: 14). Veasey (2001: 420) states that a well-defined architecture will reduce misunderstandings (miscommunication) and sharpen the focus of the enterprise. Ballengee (2010b: 50) lists communication between IT and business as one of the beneficial functions of the enterprise architecture process. This can lead to a convergence towards a common “set of models, principles, frameworks and jargon” as well as continued communication.

A study done by Kappelman *et al.* (2010: 100) found that one of the main functions of enterprise architecture is communicating the objectives of the enterprise. In a different article he also states that: “EA is about improving the ability of the people in your enterprise to communicate more quickly and effectively so they can manage and change the organisation more quickly and effectively" (Kappelman, 2010b: 120).

2.5.6.2 Strategic Governance

Included in the process of enterprise architectures, is an attempt to gain a coherent expression and implementation of strategy. This is achievable by providing a model or architecture that

- can be shared by anybody in the enterprise;
- makes use of a common, complexity-reduced language;
- includes precise strategy plans; and
- sharpens the focus on priorities (Veasey, 2001: 420).

Hoogervorst (2004: 229) argues that enterprise architecture is a key element in formally linking strategy and implementation or execution. Enterprise architecture is seen as part of the tactical phase of an enterprise – between the strategic phase and the execution phase. The planning of the implementation of strategic choices takes place in this tactical phase. Bruls *et al.* (2010: 518) state that "enterprise architecture is used to plan, govern and control the detailed architecting and engineering of individual solutions by solution architects and engineers". Enterprise architecture thus forms part of the strategic governance of an enterprise.

The correct implementation of enterprise architecture governance will improve strategic governance throughout the whole enterprise. Enterprise architecture governance includes:

- standards, guidance and metrics to the level below,
- compliance with the rules and structures of the level above, and
- communication and integration with peer functions (Baudoin, 2010: 13).

Another part of this function of enterprise architecture, is to assist the executives of an enterprise in determining the boundaries of the enterprise (O'Rourke, Fishman & Selkow, 2003: 6). It might exclude some of its departments, divide itself into different enterprises or even stretch beyond the realm of its legal domain.

2.5.6.3 Business Processes

It is a function of enterprise architecture to “[c]reate, unify, and integrate business processes across the enterprise” (Schekkerman, 2004b: 25) and optimise an enterprise’s internal operations (Jonkers *et al.*, 2006: 64). Enterprise architecture should aim to design lean and mean enterprises – “so that it is completely effective and efficient, so that it is integrated, so that it is dynamic, so that we can create new instances (implementations) on demand” (Zachman, 2010: 44).

Enterprise architecture will be “more able to succeed in a world that demands we do more with less, faster, while traditional boundaries blur, and the rules of engagement change” (Kappelman, 2010a: 36). Veasey (2001: 424) stresses that the development of architectures should be disseminated along with the strategic and business processes that uses them, otherwise resistance can be encountered. Synergy within branches should also be considered. Architecture should naturally be added to existing strategy units and processes.

2.5.6.4 Enterprise Architecture Process

“The integration of EA [Enterprise Architecture] and MDD [Model Driven Development] enables enterprises to develop a seamless workflow that begins with their enterprise architecture, takes into consideration business process analysis, and ends with application modeling and development. This integration encourages the participation of both technical and non-technical personnel in each phase of the EA and application development process, ensuring an optimal solution” (Dodani, 2008: 31).

Kluge, Dietzsch & Rosemann (2006) state that consistent and integrated methodologies are one of the intrinsic values of enterprise architecture. A holistic, end-to-end future-state architecture process will be created. This process should accurately reflect the business strategy of the enterprise (Schekkerman, 2004b: 25). Establishing the architecture also provides a basis and process to handle architectural exceptions.

A good enterprise architecture implementation also needs a “technology watch role”, to track the evolution of the enterprise architecture discipline and trends, join forums and share case studies (Baudoin, 2010: 22).

2.5.7 Reducing Risks

A somewhat less significant function of enterprise architecture is risk management (Schekkerman, 2004b: 25). If more focus is placed on risk management during the architecture process, the resulting benefits could, however, be increased substantially.

Enterprise architecture, by cleaning up the IT environment, provides the following risk-related functions:

- reduction in business risk – due to reliability and availability of systems,
- increased disaster tolerance – minimising business losses during outages,
- reduction in security breaches – for example, avoidance of computer viruses and inappropriate access to data (Ross, Weill & Robertson, 2006: 97).

In project management during a change programme, enterprise architecture assists in providing more precise scoping of the projects as well as in determining the inter-project impacts. This results in early identification of risks and subsequent avoidance of implementation failures (Veasey, 2001: 421).

2.5.8 Linking with External Partners

Schekkerman (2004b: 25) lists another benefit – an increase in the flexibility of the organisation in linking with external partners. Better quality of information and understanding of your own organisation, makes linking with suitable external organisations easier. Enterprise architecture is a valuable instrument “to get a grip on the wealth of interconnections with its customers, suppliers, and other partners” (Jonkers *et al.*, 2006: 64).

According to Baudoin (2010: 2) from the Cutter Consortium, enterprise architecture in 2010 is more about integrating external capabilities than developing in-house capabilities. The focus is on integrating all the capabilities, in order to allow IT to deliver business capabilities. Service-orientated enterprise architecture provides the ability to interoperate and collaborate with different partners. It creates a flexible environment to mix and match partners and enables the enterprise to exploit merging business opportunities quickly (Lankhorst *et al.*, 2005: 44).

2.6 DOCUMENTING ENTERPRISE ARCHITECTURE

According to Gartner (2008b: 3) documenting the enterprise architecture starts with defining the preferred future state, followed by describing the current state, with as little detail as required, and then finally the development of a roadmap. Describing the current state should result in a repository (not an inventory or configuration management database), which contains summary information. The *rationale* behind this approach is that extensive documentation of the current state is costly and time-consuming and delivers little business benefit. When new projects are executed, more detail can be added to this documentation. An annual scheduled refresh of the documentation is also advised.

Documenting the enterprise's enterprise architecture is a vital part of realising the roles and functions of enterprise architecture described above. "Because enterprise architecture is strategically driven, models describing the current target architectures should be concise and well documented to facilitate understanding of data flows in enterprise architecture" (Kaisler, Armour & Valivullah, 2005; Shah & El Kourdi, 2007). Although the enterprise architecture process can result in multiple benefits, it also has its share of challenges. Furthermore the impact of not documenting the enterprise architecture should also be taken into account. The benefits, challenges and impact of documenting enterprise architecture is describe below and is applicable, *mutatis mutandis,* to all the different domains of architecture.

2.6.1 <u>The Benefits of Documenting Enterprise Architecture</u>

"An enterprise architecture is critical for building a foundation for execution because it maps out important processes, data, and technology enabling desired levels of integration and standardization" (Ross, Weill & Robertson, 2006: 92). During the documentation and implementation process, enterprises can achieve a number of benefits. It is evident that every function (described in paragraph 2.5), if fulfilled successfully and effectively, will result in a direct benefit to the enterprise. These obvious benefits will thus not be repeated here. The benefits are, however, dependent on the efficiency and effectiveness of the specific enterprise as well as the communication accompanying the process.

Stevenson (2003) reasons that it is important for enterprises to acquire **intellectual capital** in this knowledge economy. Creating and maintaining a set of models of the enterprise (collectively called enterprise architecture) will retain a wealth of enterprise intellectual capital.

Kluge, Dietzsch & Rosemann (2006) found that, to ensure an efficient deployment, an accompanying **value** realisation process should be run together with documenting the enterprise architecture. If the value is not evident to the business, the architecture will not be fully utilised. "Quantifying the Enterprise Architecture benefits has always been a challenge because measurements and real value delivered can not often be expressed in simple technical oriented metrics only".(Schelp & Stutz, 2007: 5).

Schelp & Stutz (2007: 9) developed a balanced framework to **measure** the value of enterprise architecture, based on the balanced scorecard principles. The purpose is to provide a dashboard with information on every aspect of enterprise architecture. The value of enterprise architecture is measured on different levels within the four quadrants of services, processes, assets and finance. A set of frames have been added to the scorecard in order to limit the scope of analysis of the different metrics, within the four perspectives. The inner frame addresses the context of IT architecture, the next frame addresses the resource allocation, the third frame addresses the context of the enterprise and the outer frame indicates the context of cross-company. The framework is depicted in Figure 2-4: Enterprise Architecture Scorecard Framework (Schelp & Stutz, 2007: 9) below.

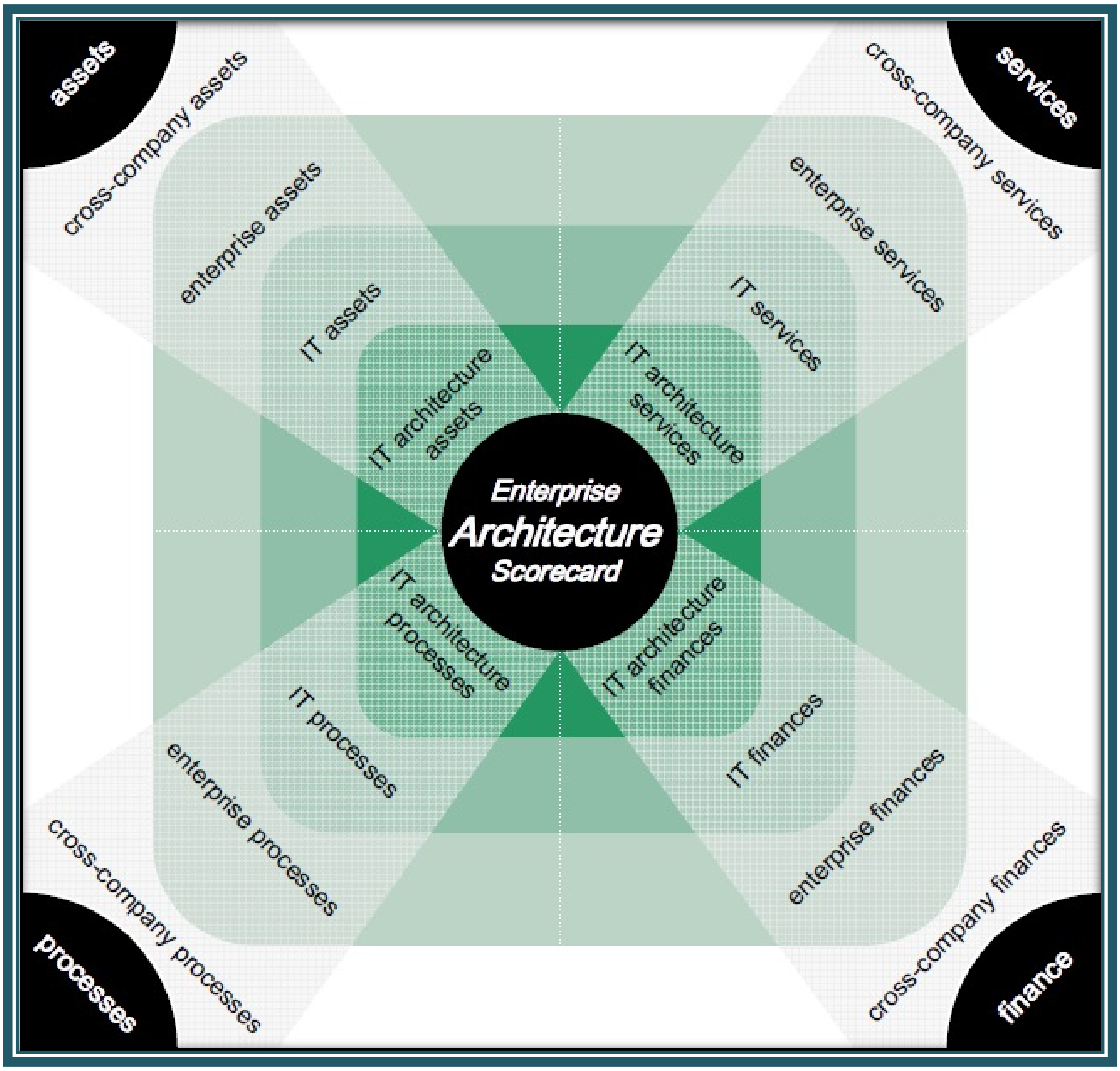


Figure 2-4: Enterprise Architecture Scorecard Framework (Schelp & Stutz, 2007: 9)

Ross, Weill & Robertson (2006: 92) found that benefits can be generated soon after an enterprise embarks on the architecture maturity journey. The benefits keep growing as the enterprise moves into later architecture stages and can multiply, as long as the enterprise keeps on learning. Although the rate of change in IT is believed to be quite fast, architectures have rather long lives. “They take several years to establish and can often expect to be useful for ten years and more” (Veasey, 2001: 420). The Seven Layer Model from Open Systems Interconnect is a good example of this. The benefits can thus start very early in the process and exist or grow over a long period (ISO, 1996).

2.6.2 The Challenges and Problems of Documenting Enterprise Architecture

The process of documenting enterprise architecture has several challenges. Zachman (1996: 5) – the father of enterprise architecture – describes it as follows: "It makes little difference whether the object is physical, like an airplane, or conceptual, like an Enterprise. The challenges are the same. How do you design and build it piece-by-piece such that it achieves its purpose without dissipating its value and raising its cost by optimizing the pieces, sub-optimizing the object."

"Enterprise architecture addresses the **double challenge** of increasing IT efficiency while continuing business innovation" (Shah & El Kourdi, 2007: 36). Wegmann (2003: 4) discusses another challenge for enterprise architecture: to deal simultaneously with complicated systems (deterministic) and complex systems (non-deterministic and including people). "Architects and specialists are well trained for dealing with complicated systems but are usually far less comfortable with complex systems". Combining these in an overarching architecture is the challenge. Shah & El Kourdi (2007) see this as two perspectives, namely: enterprise architecture frameworks and the organisation structure.

However, not every enterprise architecture will provide the desired outcomes. Innovative design in the creation of the architecture is essential to ensure that it is adaptable and contains within itself the ability to change. Such a design requires a high degree of skill, experience and dedication on the part of the designers (Stevenson, 2003). The **scalability** of enterprise architecture is part of this challenge. This is especially true for models with different underlying repositories and dependencies (Shah & El Kourdi, 2007: 41).

The lack of **measurements** and assessment tools provides a challenge to advocating and implementing enterprise architecture (Schelp & Stutz, 2007; Shah & El Kourdi, 2007: 40). The metrics and assessment tools should be designed early in the enterprise architecture process in order to mitigate this risk.

The different models or tools, used by different stakeholders, may reflect the stakeholders limited perspective and may easily result in **ambiguous documentation**, instead of a big picture, represented in a homogenous way (Shah & El Kourdi, 2007: 40).

Due to the dynamic nature of the competitive environment (Lin & Dyck, 2010) it is a challenge to accommodate the **rate of change** in both the technology and the businesses side.

Kaisler, Armour & Valivullah (2005: 2) lists **maintenance** of models and documentation as a critical challenge. "Maintenance is essential to an EA because operational consistency must be preserved while the organization continues to evolve the architecture". Maintenance has to be attended to, while in the meantime normal day-to-day operations cannot be impacted upon. This creates an on-going tension between operations and new systems (Shah & El Kourdi, 2007: 40).

The co-operation and current state of the **enterprise** can be a challenge. Implementation of enterprise architecture can be tedious and create uncomfortable stakeholders within the enterprise (Shah & El Kourdi, 2007). During a case study, Thundatil (2007: 469) found it challenging to implement an architecture due to: insufficient existing data; lacking processes; and the lack of technology infrastructure. The lack of integration across enterprise functions can add to this problem. Another issue is the challenge of using the architecture model efficiently across the whole enterprise, including all the cross-interactions between different entities (Shah & El Kourdi, 2007: 40).

There are still many challenges in **communicating** and promoting enterprise architecture and educating the enterprise about enterprise architecture – this stems from the fact that it is not yet mature in all aspects (Wegmann, 2003: 1). It also presents a challenge to have all project teams and entities within an enterprise informed and adhering to the enterprise architecture principles. It thus follows logically that the **governance** of enterprise architecture is a challenge that has to be managed (Kaisler, Armour & Valivullah, 2005: 1; Schelp & Stutz, 2007: 7).

It is evident that the enterprise architecture process is a challenging one. The challenges vary from dealing with IT efficiency versus business innovation, scalability, measuring, creating unambiguous documentation, rate of change, maintaining models, co-operation/readiness of enterprise and enterprise communication to governance.

2.6.3 The Impact of not Documenting Enterprise Architecture

Documentation of enterprise architecture can be done on different levels of abstraction. It is important to obtain the best level of documentation for the specific enterprise. If the documentation is lacking or not done at all, the following impacts have been deduced from the literature and should be considered:

- loss of intellectual capital, knowledge and concepts with staff turnover, which may never be recovered or be very expensive to rebuild;
- limitation on enterprise wide communication, as no documentation is available to distribute to stakeholders for awareness, input or information;
- inability to gain official approval from management (Kappelman, 2010c: 14);
- incomplete architecture, as it is nearly impossible, due to complexity, to create a comprehensive architecture in a large enterprise without being documented;
- difficulty in measuring the worth of enterprise architecture;
- lack of consistent understanding and implementation across the entire enterprise (Kappelman, 2010c: 14);
- increased systems costs and sub-system problems (Long, 2008: 62);
- inability to govern the enterprise architecture process; and
- lack of maturity in enterprise architecture.

2.7 EXISTING FRAMEWORKS

The dictionary defines a framework as 1) “a structure for supporting or enclosing something else, especially a skeletal support used as the basis for something being constructed”; 2) “a set of assumptions, concepts, values, and practices that constitutes a way of viewing reality” (Webster's New World Dictionary, 2011). In the context of enterprise architecture it indicates “a model or outline that provides the logical structure within which EA deliverables will be created and related to each other” (Westbrock, 2007: 6) for providing a simplified context for the scope and structure of an enterprise and its components. A framework has several elements such as “methodology, product descriptions, reference models, categorization and classification" (Long, 2008: 140). Examples are Zachman Framework, TOGAF, DoDAF, MODAF (Ministry of Defence Architecture Framework), FEAF and Ontology (Zachman, 2008).

Frameworks have a twofold role, namely as tools for documenting and specifying different components and as tools for planning solving problems. These roles are better explained in Figure 2-5.

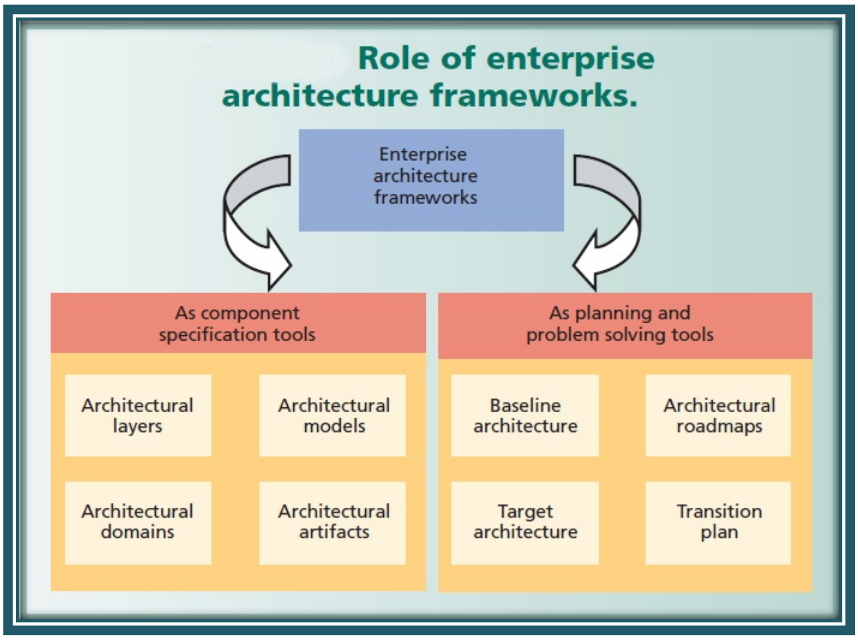


Figure 2-5: Role of Frameworks (Shah & El Kourdi, 2007: 38)

The Department of National Defence and Canadian Forces (2009: 7) perceives a framework as the graphs, models and narratives used in a unified way, to describe the architecture design. This is necessary to achieve a standard medium for communicating an explicit and repeatable presentation of the various views of the architecture tailored to the specific enterprise. A framework is one of the first steps in enterprise architecture but does not include information about the implementation process.

According to TOGAF's definition: (The Open Group, 2009c: 7) an "architecture framework is a foundational structure, or set of structures, which can be used for developing a broad range of different architectures." It should also describe a method, a set of tools, provide a common vocabulary and include a list of recommended standards and compliant products that can be used to implement the building blocks.

A framework assists in organising the thinking, in providing descriptions of artefacts, in using a common semantics, in communicating, and in indicating relationships and coherence between the elements (Abdallah & Galal-Edeen, 2006: 1). Frameworks are essential in enterprise architecture because of the need of providing a simplified presentation of the complexity, organising the multitude of entities and relations, providing highlights and utilising known practices (Westbrock, 2007: 8). “An Enterprise Architecture Framework (EAF) maps all of the software development processes within the enterprise and how they relate and interact to fulfil the enterprise’s mission” (Urbaczewski & Mrdalj, 2006: 18). A framework should be broadly usable for capturing information in the enterprise and provide a mechanism to access, organise and display this information (Urbaczewski & Mrdalj, 2006: 19).

Schekkerman (2004a: 16) came to the conclusion that the existing frameworks have different evolutions, serve different purposes, are based on different principles, are different in scope and approach and have different structures.

Research (Rosen, 2010: 2) showed that TOGAF, DoDAF and Zachman Frameworks are being actively used in enterprises. A summarised overview of a selection of frameworks, *i.e.* the important and currently used frameworks, follows below. Each framework will be described in terms of overview, scope, views, abstractions, Systems Development Life Cycle components, strengths and weaknesses.

2.7.1 Zachman Framework

The first framework to be described is the Zachman Framework, which has been evolving since 1987. Although self-described as a framework, according to Sessions (2007) it is actually a taxonomy. It will be described according to the above-mentioned headings.

Overview: The Zachman Framework is a schema indicating the intersection between the primitive interrogatives (what, how, where, who, when and why) and the reification transformations from an abstract idea to an instantiation (identification, definition, representation, specification, configuration and instantiation). This is typically displayed as a six by six bounded matrix, as illustrated in Figure 2-6 (Zachman, 2008).

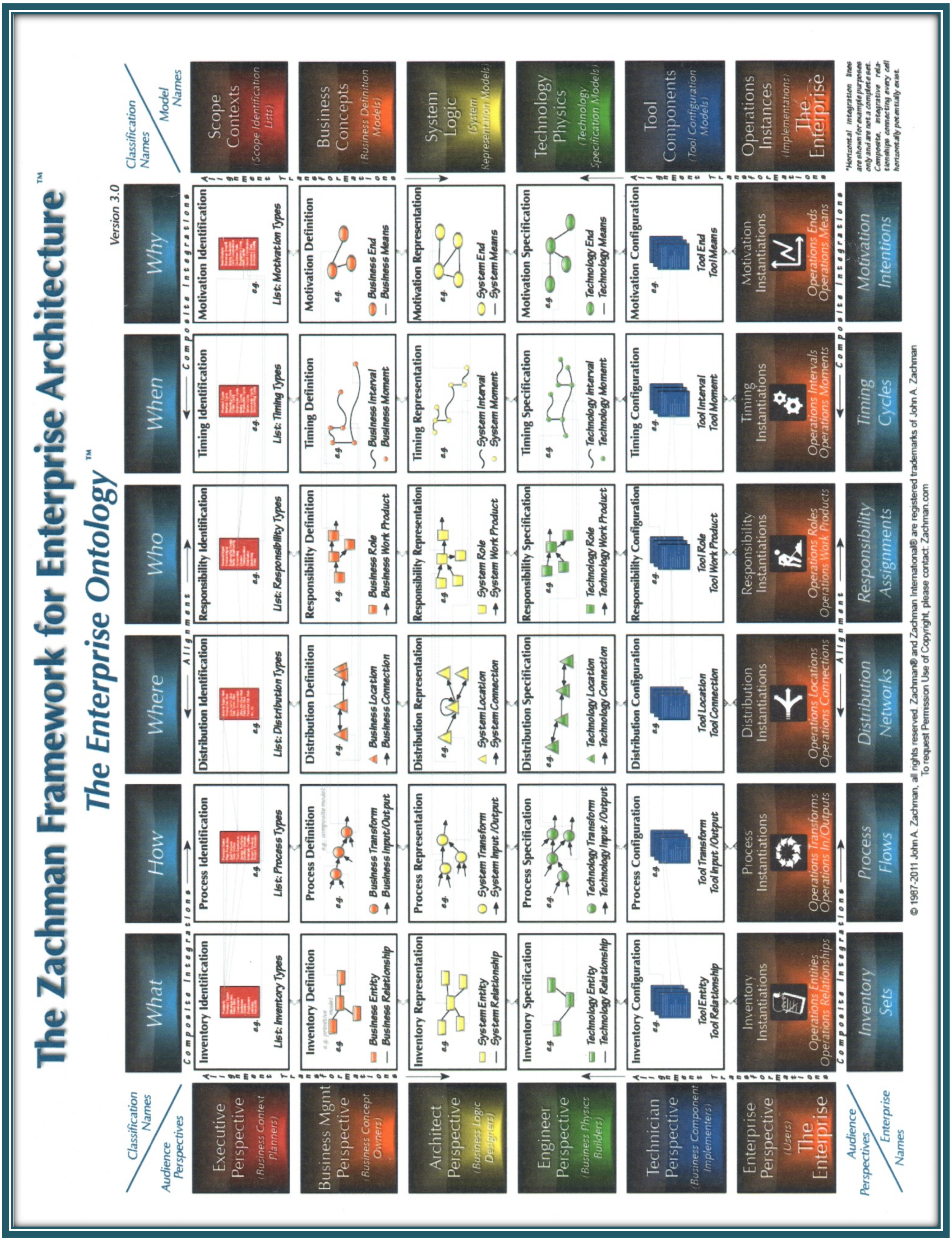


Figure 2-6: The Zachman Enterprise Framework (Zachman, 2011)

Scope: The Zachman Framework is an ontology (structure) for describing the enterprise, and is not a methodology for creating the implementation. It can also be described as a metamodel (Zachman, 2008).

Views: The Zachman Framework covers the following views (called perspectives), based on complex industrial products (Zachman, 2010: 40):

- Strategists – Scope (Boundaries) – Scope
- Owners (Executives) – Requirements (Concepts) – Business
- Designers (Architects) – Schematics (Logic) – System
- Builders (Engineers) – Blueprints (Specifications)– Technology
- Implementers (Technicians) – Tooling (Configurations) – Component
- Operators (Workers) – Implementation Instances – Operations.

Abstractions: The Zachman Framework covers the following abstractions, based on complex industrial products:

- What – Bill of Material – Inventory
- How – Functional Specifications – Process
- Where – Drawings – Network
- Who – Operating Instructions – Organisations
- When – Timing Diagrams – Timing
- Why – Design Objectives – Motivation (Zachman, 2010: 38).

Systems Development Life Cycle Components: The framework addresses the planning, analysis, design and implementation phases of the life cycle (Urbaczewski & Mrdalj, 2006: 22).

Strengths:

- The framework provides a classification for describing artefacts that are comprehensible and simple (Zachman, 1996: 138). It can also be used to sharpen the focus of the different artefacts (Sessions, 2007).
- The framework is comprehensive and provides input opportunities for all the stakeholders (Sessions, 2007)
- The framework is neutral to tools and methodology.
- The framework is a planning tool, which helps to put the initiatives in context (Zachman, 1996: 5).

- Focused attention can be paid to a specific intersection and keeping the bigger picture in mind or in other words, it separates the level of detail (Westbrock, 2007: 14).
- It is rigorous and useful for guiding strategists and IT departments (Vail, 2002: 8).

Weaknesses:

- It does not describe the process or methodology (Vail, 2002: 9).
- It is unclear which cells are more important, as nobody fills in all the cells (Sessions, 2008: 4).
- “Can lead an EA practitioner to focus on current state and near term to the detriment of the To-Be, Future State, Target EA” (Westbrock, 2007: 14).
- It is a single cell model – that is a primitive model.
- It does not address the maintenance phase of the systems development life cycle (Urbaczewski & Mrdalj, 2006: 22).

2.7.2 TOGAF

The Open Group Architecture Framework divides enterprise architecture into four divisions, namely business architecture, application architecture, information architecture and technical architecture – Figure 2-7. The “most important part of TOGAF is the Architecture Development Method, better known as ADM” (Sessions, 2007). The description of TOGAF follows. Detailed descriptions of every component and process of TOGAF are freely available.

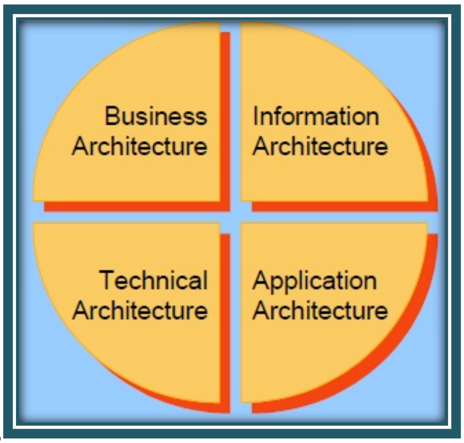


Figure 2-7: TOGAF Architectures (Morar, 2008: 7)

Overview: TOGAF uses a continuum of architectures, from generic to specific. It starts with *Foundation Architectures*, followed by *Common Systems Architectures, Industry*

Architectures and ends with *Organizational Architectures*. This continuum can be applied to the enterprise, the architectures and the solutions. The TOGAF ADM consists of eight phases in a cycle, namely architecture vision, business architecture, information systems architectures, technology architectures, opportunities & solutions, migration planning, implementation governance, architecture change management – see Figure 2-8. Besides addressing the different architectures and providing an architecture process, it also provides a framework for architecture governance – see Figure 2-9. Version 9 also contains the Content Metamodel for traceability of artefacts. Other components are the Technical Reference Model (TRM), providing a taxonomy for IT infrastructure components and the Standards Information Base (SIB) (The Open Group, 2009a; Westbrock, 2007: 17).

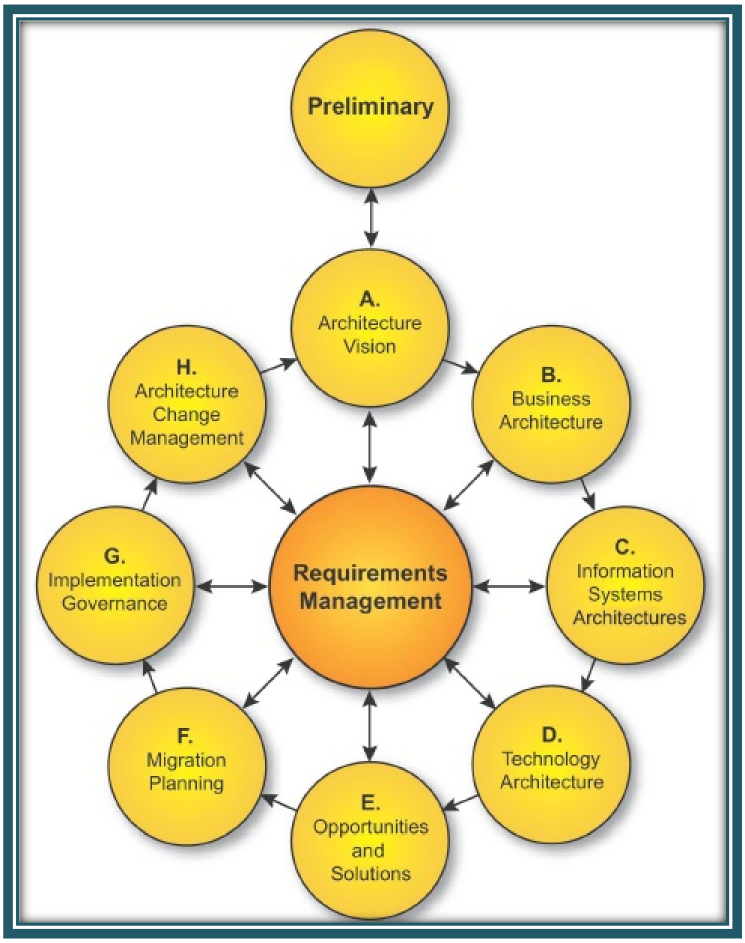


Figure 2-8: TOGAF ADM Phases (The Open Group, 2009c: 54)

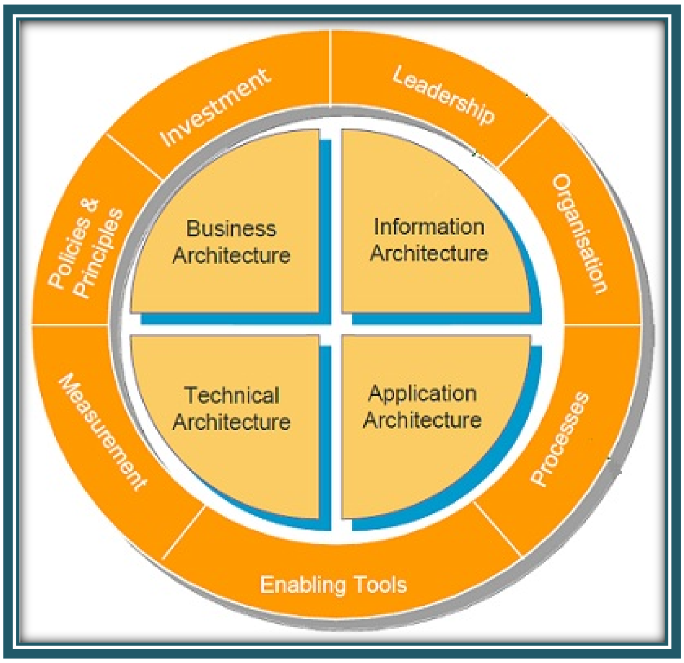


Figure 2-9: TOGAF Governance (Morar, 2008)

Scope: TOGAF is a framework, containing a detailed method, a set of supporting tools for developing an enterprise architecture as well as a set of recommended standards (Morar, 2008: 4).

Views: According to Urbaczewski & Mrdalj (2006: 20) TOGAF provides only the Owner, Designer and Builder views.

Abstractions: TOGAF addresses the How and Who abstractions (Urbaczewski & Mrdalj, 2006: 21).

Systems Development Life Cycle Components: As TOGAF contains **"**principles that support decision-making across enterprise; provide guidance of IT resources; support architecture principles for design and implementation" (Urbaczewski & Mrdalj, 2006: 22), it addresses the Analysis, Design and Implementation phases of the Life Cycle.

Strengths:

- The framework is neutral to vendors and technology (The Open Group, 2009a).
- The framework is process-driven and not artefact-driven (Morar, 2008: 5).
- All documentation for TOGAF is published online and has been downloaded extensively (Gerber, Kotzé & Van der Merwe, 2010: 57).
- It is generic and flexible (Morar, 2008: 5) and can be adapted to work with other frameworks and approaches (Westbrock, 2007: 23).
- It provides a set of conceptual tools (Morar, 2008: 5).
- Training is available for certification (Westbrock, 2007: 23).

- It is possible to co-implement with ITIL (Information Technology Infrastructure Library) and an integration model exists (Thorn, 2007).

Weaknesses:

- There are lots of detail in the framework (Westbrock, 2007: 23), which can hamper understanding and implementation.
- The process is long and tortuous (Sessions, 2008: 6).
- It can "lead startup efforts into too much too soon" instead of "just enough architecture, just in time" (Westbrock, 2007).
- It contains no prescriptive templates for drawing up artefacts (Sessions, 2008: 6).

2.7.3 DoDAF

The Department of Defence of the United States of America was in need of a consistent architecture, due to the compliance requirements as well as the size and complexity of their enterprise. "DoDAF version 1.5 is a transitional version as a response to the DoD's migration towards NCW" (Net-Centric Warfare) (USA. Department of Defence, 2007a: 1).

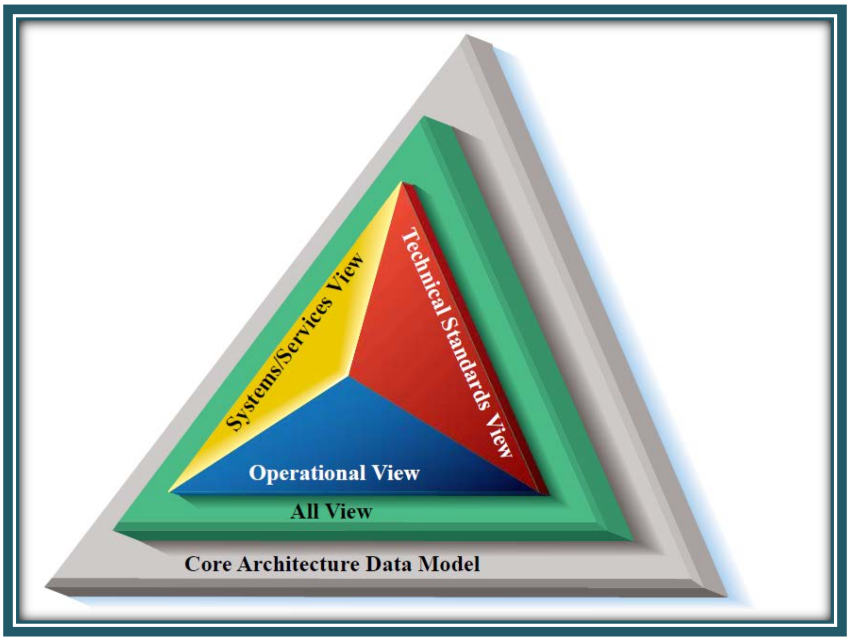


Figure 2-10: DoDAF (USA. Department of Defence, 2007b)

Overview: The DoDAF makes use of three views, namely: the operational view, the systems and services view and the technical standards view as in Figure 2-10. In the

"All view" (see Figure 2-11) these views are linked with each other. The data layer is captured in the CADM (Core Architecture Data Model) to support the exchange of information. The All View (AV) provides "information pertinent to the entire architecture" but does not "represent a distinct view of the architecture. AV products set the scope and context of the architecture" (USA. Department of Defence, 2007a: 1-9). It has detailed descriptions for the whole set of artefacts and rules for consistency. All the artefacts and architectures are stored in the DARS (DoD Architecture Registry System). DoDAF also contains a six-step process for building an architecture description (USA. Department of Defence, 2007a: 2-2).

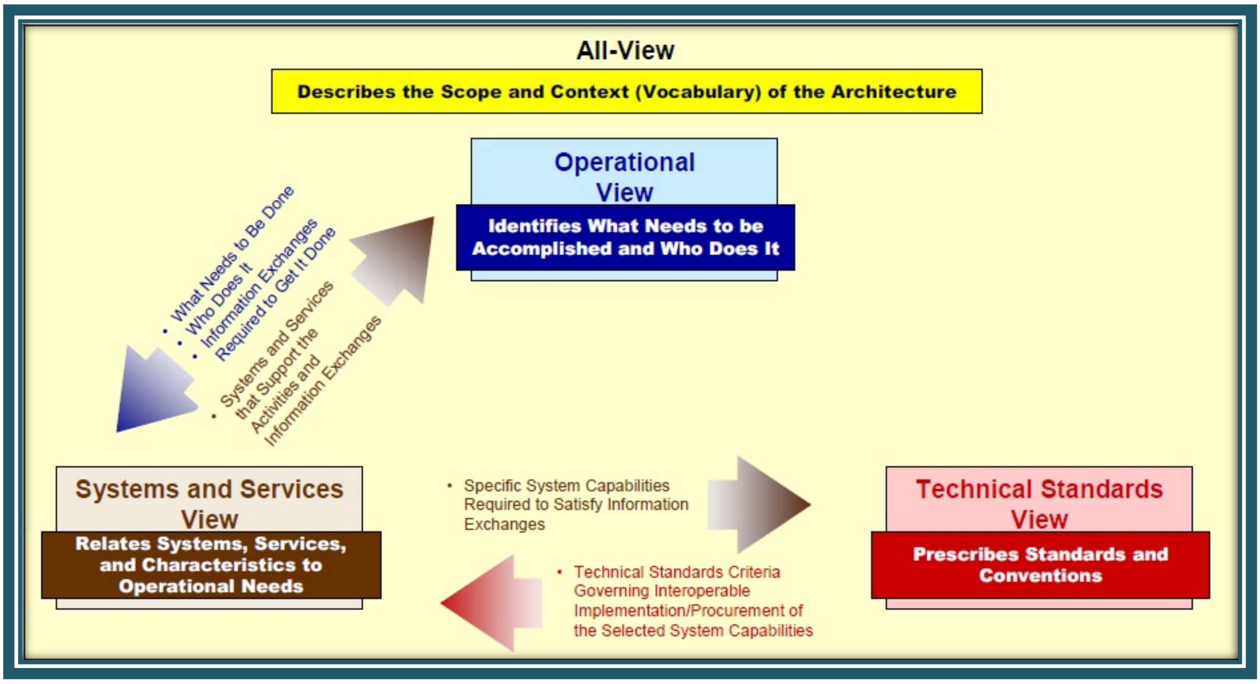


Figure 2-11: DoDAF links between Views (USA. Department of Defence, 2007a: 1-8).

Scope: The DoDAF is "a guide for the development of architectures". It provides guidance and rules based on a common denominator across the bigger department. It "is intended to ensure that the architecture descriptions can be compared and related" throughout the DoD (USA. Department of Defence, 2007a: 1).

Views: The following views are addressed (Urbaczewski & Mrdalj, 2006: 20):

- Strategists/Planner – All View,
- Owners (Executives) – Operational View,
- Designers (Architects) – Systems View,
- Builders (Engineers) – Technical View.

Abstractions:

- Operational (operational nodes, tasks and activities),
- Systems and Services (system, service and interconnection functionality), and
- Technical Standards (minimal set of rules governing the arrangement, interaction and interdependence of system elements) (USA. Department of Defence, 2007a: 1-8).

Systems Development Life Cycle Components: The framework addresses the planning, analysis and design phases of the life cycle as well as a part of the implementation phase with the descriptions of the final products (Urbaczewski & Mrdalj, 2006: 22).

Strengths:

- Detailed descriptions of framework and artefacts exist.
- It follows a structured, standardised and logical approach.
- It has been accepted as a standard by the DoD communities.
- It is a "mature framework for arch modeling of a system of systems".
- It has been accepted by industry and systems engineering professional and standard societies (USA. Department of Defence, 2005: 30).
- "A unique strength of DoDAF is the requirement of a glossary as a top-level artifact in describing the architecture of a system" (Thario, 2008).

Weaknesses:

- It does not address the time (when) and motivation (why) abstractions (Urbaczewski & Mrdalj, 2006: 21).
- It has a rather specific target (Lankhorst *et al.*, 2005: 29).
- It is not capability focused – it is difficult to show impact of decisions.
- There is not enough integration, action timing and sequencing between products.
- The diagrams are too complex to present to senior leadership.
- It encourages "stovepipe" services (USA. Department of Defence, 2005: 31).

2.7.4 FEAF

FEAF was commissioned by the Office of Management and Budget of the Executive Office of the President as part of FEA to align IT support with the business functions, to manage IT spending and identify possibilities for the re-deployment of IT assets in the

Federal government of the USA (USA. FEA Program Management Office, 2004: 2). “The FEAF provides a structured approach to federal organizations to integrate their respective architectures into the federal enterprise architecture (FEA)” (Saha, 2007: 3).

Overview: FEAF consists of a set of interrelated reference models, namely: the Business Reference Model (BRM), Service Component Reference Model (SRM), Performance Reference Model (PRM), Data Reference Model (DRM) and Technical Reference Model (TRM). It follows a business-driven approach (see Figure 2-12). FEA also aims to use a common process view, to help sections to discover similar processes.

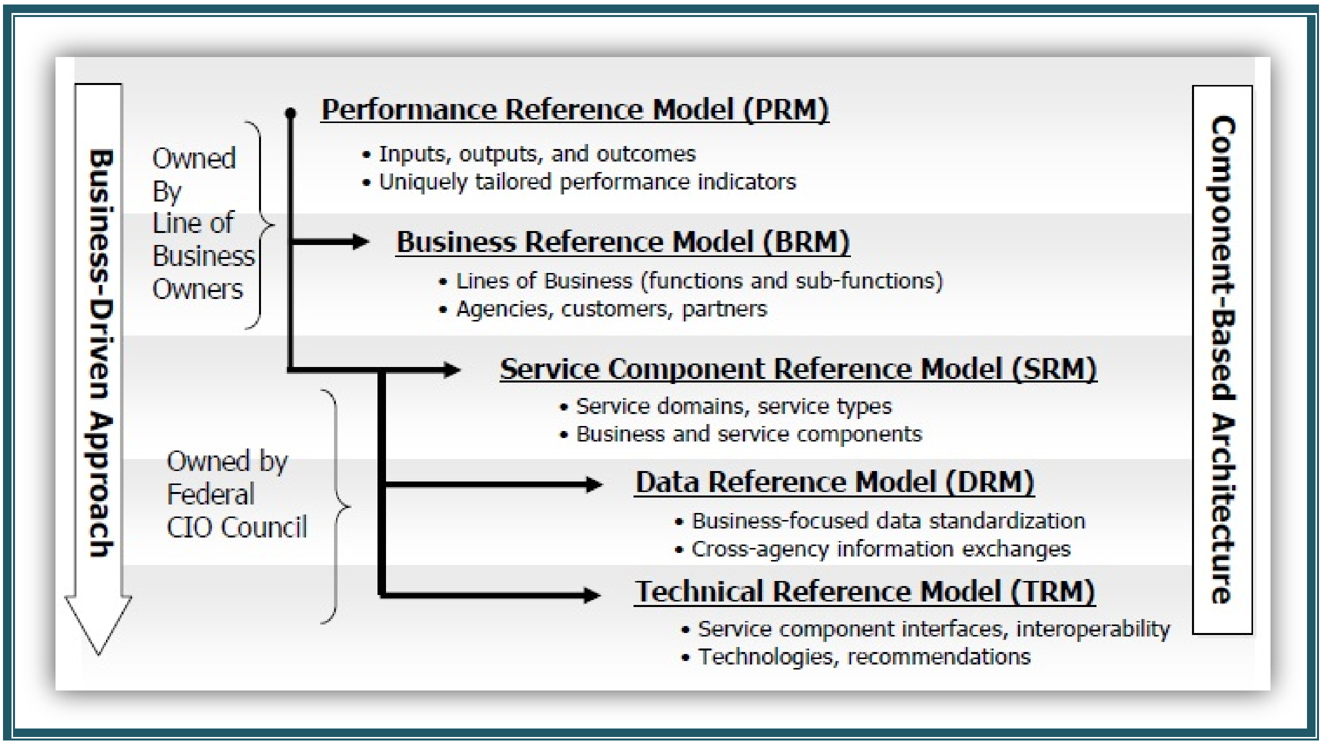


Figure 2-12: FEAF (USA. FEA Program Management Office, 2004: 7)

Scope: “The FEA consists of a set of interrelated ‘reference models’ designed to facilitate cross-agency analysis and the identification of duplicative investments, gaps and opportunities for collaboration within and across agencies. Collectively, the reference models comprise a framework for describing important elements of the FEA in a common and consistent way” (Executive Office of the President of the United States, 2007: 5). FEAF encompasses all aspects like business architecture, data architecture, technology architecture and application architecture (Saha, 2007: 3), see Figure 2-13. The FEA Practice subscribes to three core principles, namely:

- Business-driven: aligning to the government’s strategic plans;

- Proactive and collaborative across the federal government; and
- Architecture improves the effectiveness and efficiency of government information resources (Executive Office of the President of the United States, 2007: 4).

	Data Architecture	Applications Architecture	Technology Architecture
Planner Perspective	List of Business Objects	List of Business Processes	List of Business Locations
Owner Perspective	Semantic Model	Business Process Model	Business Logistics System
Designer Perspective	Logistic Data Model	Applications Architecture	System Geographic Deployment Architecture
Builder Perspective	Physical Data Model	Systems Design	Technology Architecture
Subcontractor Perspective	Data Directory	Programs	Network Architecture

Figure 2-13: FEAF Matrix (Other Architecture and Frameworks, 2006)

Views: FEAF recommends five views, namely the planner's, owner's, designer's, builder's and subcontractor's view (Saha, 2007: 6; Urbaczewski & Mrdalj, 2006: 20).

Abstractions: The following abstractions are covered by FEAF:

- What – Data Architecture – Entities;
- How – Applications Architecture – Activities;
- Where – Technology Architecture – Locations (Urbaczewski & Mrdalj, 2006: 21).

Systems Development Life Cycle Components: It is deduced from Saha that FEAF covers the planning, analysis, design and implementation phases of the life cycle as well as maintenance (Saha, 2007: 4; Urbaczewski & Mrdalj, 2006: 22).

Strengths:

- FEAF does not prescribe a specific set of tools (Saha, 2007: 12).
- FEAF is very widely implemented and not only by the government.
- it is a comprehensive and full-fledged methodology,
- has a wealth of best practices and guidelines, and
- is non-industry specific (Saha, 2007: 2).

Weaknesses:

- It lacks "a single unified architecture development language based on a common underlying metamodel",
- and is deficient in providing limited guidance and specifications (Saha, 2007: 16).

- It does not provide a full methodology process for creating enterprise architecture,
- and it can take a lot of time to gain value from using FEAF (Sessions, 2007).

2.7.5 Other Frameworks

There are several other lesser used or published frameworks. The E2AF, SAGA and GERAM frameworks will be discussed even more briefly below.

2.7.5.1 E2AF

The E2AF (Extended Enterprise Architecture Framework) from the Institute For Enterprise Architecture Developments based (IFEAD) in the Netherlands supports communication and collaboration with all the relevant extended stakeholders by dealing with the processes and activities of extending enterprise architecture beyond its original and enterprise boundaries. E2A has a clear structure with powerful implications. It consists of a four by six matrix, which addresses six abstractions (contextual, environmental, conceptual, logical, physical and transformational level) and the four rows addressing the business, information, information systems and technology/infrastructure aspects (Schekkerman, 2004b: 91-98).

2.7.5.2 SAGA

SAGA (Standards and Architectures for eGovernment Applications) was established by the German Federal Government to make progress towards modern and service-orientated administration. The purpose is to ensure information flow, establish comparable standards, provide specifications and ensure applicability. It defines three target groups, *i.e.* government to citizens, government to business and government to government. Within these columns (areas) more than 350 services along the value chain was analysed to enable eight service types. Furthermore the five viewpoints of the Reference Model for Open Distributed Processing (RM-ODP) were also adopted for SAGA (Schekkerman, 2004b: 191-196).

2.7.5.3 GERAM

The aim of GERAM (Generalized Enterprise Architecture and Methodology) is to generalise the common requirements of the various enterprise architectures. It is not a framework as such, but a methodology to classify other frameworks and their artefact types (Saha, 2007: 2). GERAM “defines a tool-kit of concepts for designing and maintaining enterprises for their entire life-history”. (IFIP–IFAC Task Force on Architectures for Enterprise Integration, 1999: 4). In order to gain maximum benefits from the process, GERAM provides a description of all the elements recommended in enterprise engineering and integration and thus sets a standard for the collection of tools and methods. It consists of Generic Enterprise Reference Architecture (GERA), Enterprise Engineering Methodology (EEM), Enterprise Modelling Languages (EML), Generic Enterprise Modelling Concepts (GEMC), Partial Enterprise Models (PEM), Enterprise Engineering Tools (EET), Enterprise Models (EM) and Enterprise Operational Systems (EOS) (IFIP–IFAC Task Force on Architectures for Enterprise Integration, 1999: 6).

2.7.5.4 WSDOT Architecture Reference Model

The Washington State, Department of Transport utilises the so called WSDOT architectural reference model, as depicted in Figure 2-14.

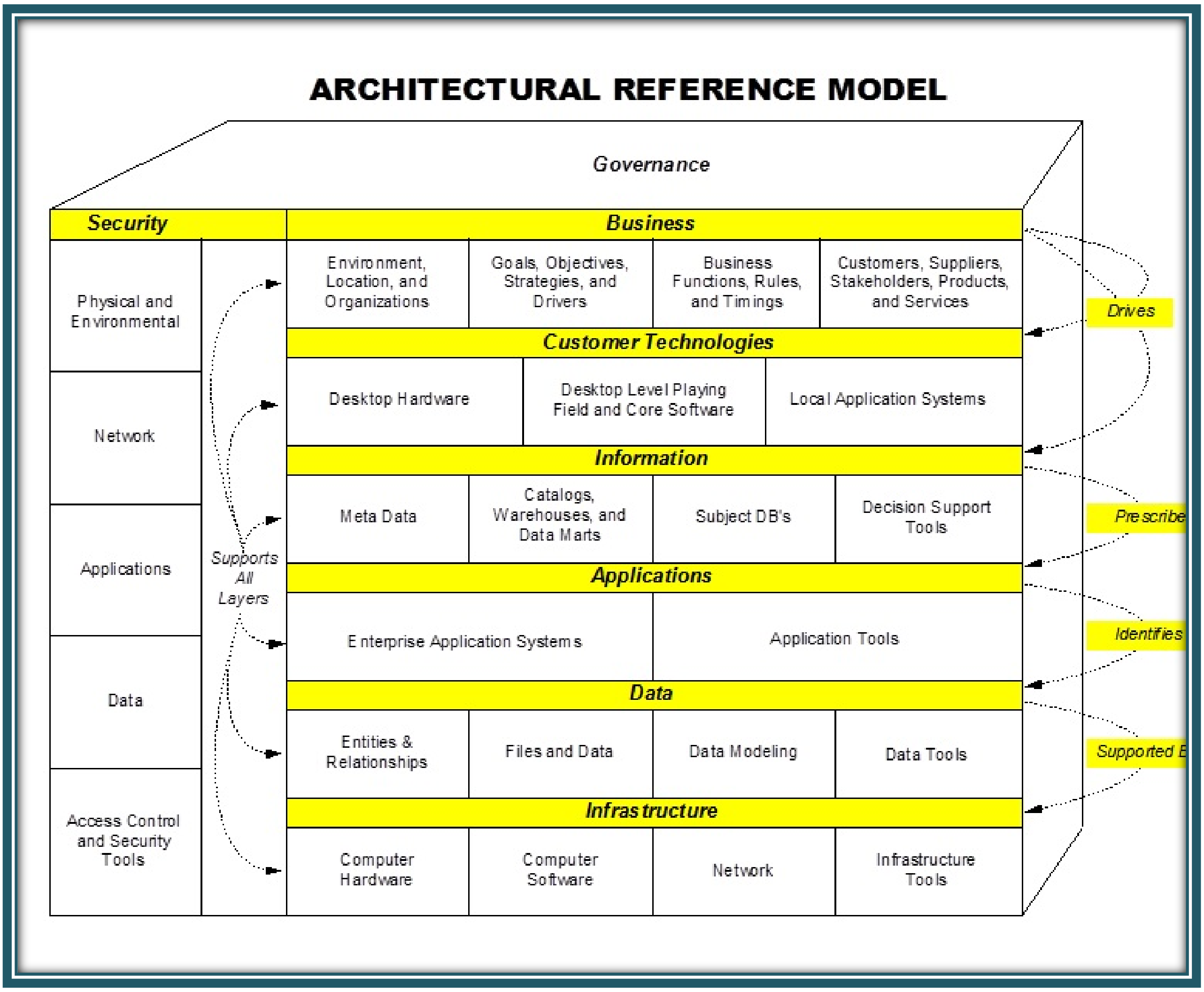


Figure 2-14: WSDOT Architectural Reference Model (Washington State. Department of Transport, 2001: 29)

This framework "is a way of representing the different sets of related architectural layers that make up the overall architecture." Each layer has associated components, which can be broken down into more detail. The WSDOT domains are divided into:

- Governance layer – wraps around all the technical layers and forms a basis for logical consistency across all the layers;
- Business layer – business-driven strategies, requirements and trends to establish the requirements for the technical architecture;
- Customer technologies – technologies at the desktop of the business person;
- Information – set of standards/rules for the programs, commands and interfaces associated with the generation of information;
- Applications – purchase, development, enhancement, delivery and support of business application software;

- Data – a consistent and universal representation of important elements;
- Infrastructure – the computing environment, including the logical and physical elements, carriers, protocols, platforms, operating systems, database environments and networks;
- Security – a vertical layer to support and protect confidentiality, integrity, privacy and recoverability of physical facilities, networks, applications and data (Washington State. Department of Transport, 2001: 30).

2.8 EXISTING MODELS

The dictionary defines a model as: “a small copy or imitation of an existing object, as a ship, building, *etc*., made to scale; a preliminary representation of something, serving as the plan from which the final, usually larger, object is to be constructed” as well as “to base your method of behavior on another person or procedure” (Webster's New World Dictionary, 2011) and “…a three-dimensional representation of a person or thing or of a proposed structure, typically on a smaller scale than the original…” as well as “a thing used as an example to follow or imitate” (Oxford Dictionaries, 2011).

The use and interpretation of the term ‘model’ within the enterprise architecture context are not consistent, for example:

- Some authors regard a model and a framework as the same concept, for example: ”A Framework for an EA is a model used by an organization to develop good corporate governance, creating added value for their business.” (Alonso, Verdún & Caro, 2010). They also list Zachman, FEAF and TOGAF as models.
- Bedwell uses frameworks and methods as inputs to model the enterprise architecture (2006: 2). He models the as-is status, the interim status as well as the to-be status using a chosen framework.
- A model represents the fundamental organisation of a system (Fischer, Winter & Aier, 2010: 193).
- Vail (2002: 8) states that an architecture consists of a set of aligned models, and also describes the Zachman Framework as a model.
- Abdallah & Galal-Edeen (2006: 5) describe the purpose of architectural models as to “provide consistent standards to document architecture specifications for the

planning, management, communication and execution of activities related to system development".

- The frameworks, like TOGAF and DoDAF, provide detail descriptions of different models to be used for different parts of the enterprise architecture.

These conflicting uses of the term 'model' highlight the problem of differentiating between frameworks and models, and the problem of integrating or separating them, combined with the different possible levels of a model/framework. For the purpose of this research it was decided to describe frameworks and models separately, with frameworks providing the outlining structure of the architecture and models being used for filling in the information within the different sections of the framework.

Generic models can be utilised to describe or design specific artefacts as part of the enterprise architecture process. A short description will be provided of a few generic models, which could be utilised for enterprise architecture.

2.8.1 <u>Process Model</u>

One of the possible generic models is a process model. There are several industry standards for modelling processes, a few examples are BPMI (Business Process Management Initiatives), BPMM (Business Process Maturity Model), BPML (Business Process Modelling Language), WSCL (Web Services Communication Language), BPDM (Business Process Definition Metamodel) and WfMC (Workflow Management Coalition) (Anon., 2008a). Process modelling is independent of the organisational structure and an example is shown in Figure 2-15. The example shows the core, supporting, planning and governance processes. Overlays for organisational structure, projects (requests and actual) or supported applications can be superimposed over the process model (Bedwell, 2006: 17-21).

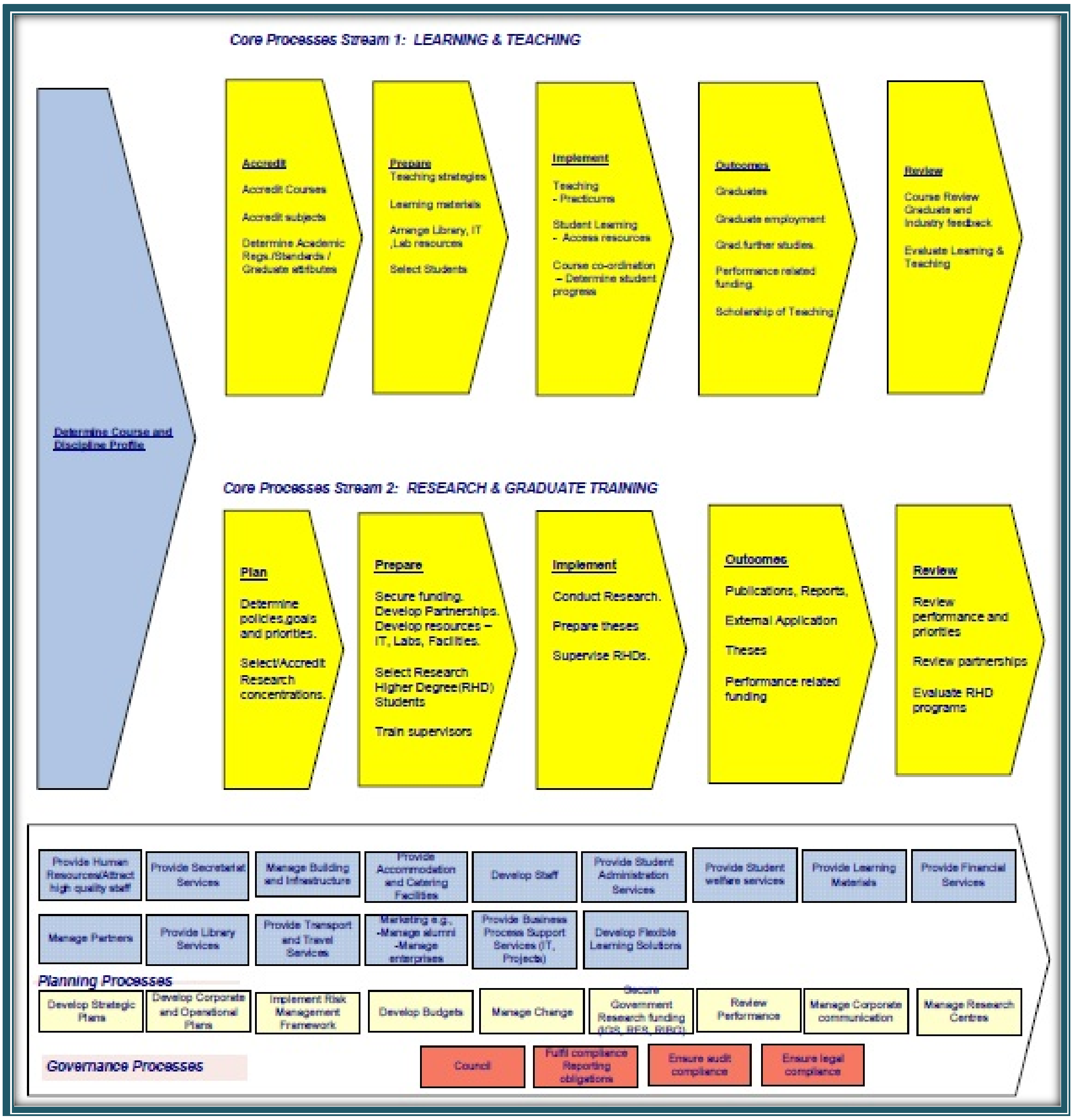


Figure 2-15: Example of Process Model (Bedwell, 2006: 17)

2.8.2 <u>Relational Model</u>

Another possible model is a relational model. The relational model is a structured way of documenting the pieces of information in relation towards each other, which supports simple, powerful querying of data (Ramakrishnan, s.a.: 14). The benefit of this is the ability to see who, how, when, where and what is affected, when something is changed (Bedwell, 2006: 29). An example is displayed in Figure 2-16.

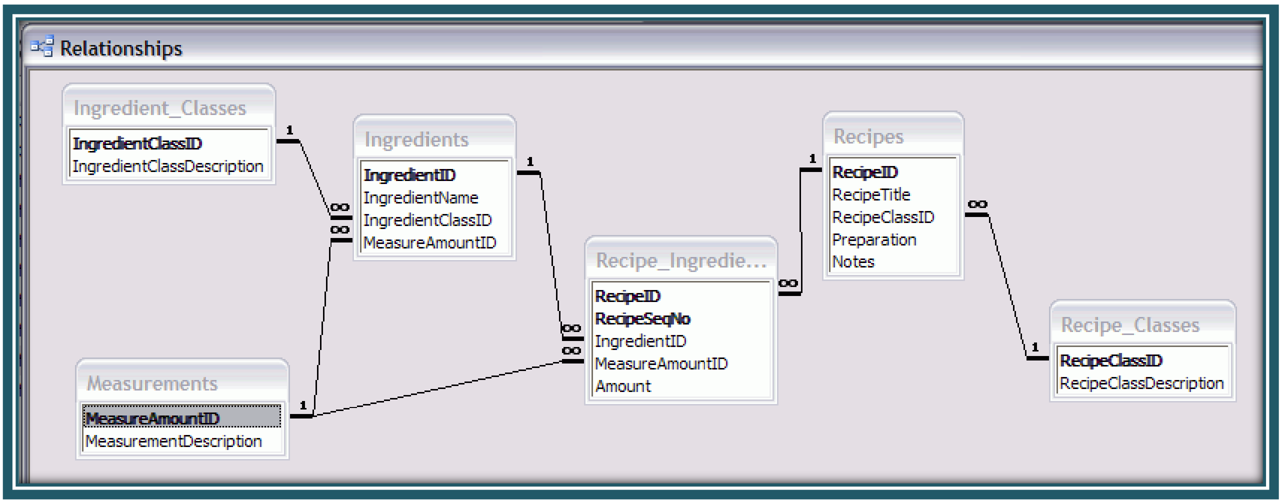


Figure 2-16: Example of Relational Model (*Experts Exchange*, *s.a.*)

2.8.3 Causal Loop Diagram Model

Causal Loop Diagrams (CLD) originated from the Systems Thinking discipline (Gharajedaghi, 2006). It depicts the cause and effect of relationships between systems of related variables. A simple example is visible in Figure 2-17. Relationships are recorded in terms of directions and time delays, and will show simultaneous causal relationships. “CLD methodology similarly captures and integrates key variables across different components of the enterprise, from strategies to processes and IT capabilities” (Vail, 2002: 10).

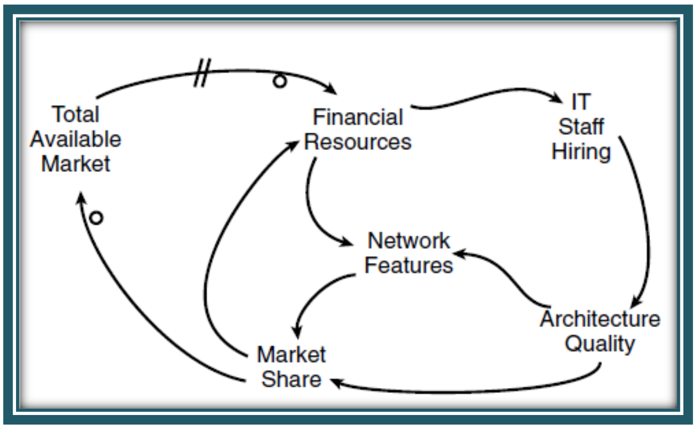


Figure 2-17: Example of a CLD (Vail, 2002: 10)

2.8.4 Object-Oriented Model

The object-oriented model contains objects (nearly anything can be an object) within objects to an arbitrarily deep level of nesting, organised into classes, where each object has its own unique identity, independent of the values it contains (Zaïane, 1995). Each class can have parents and children with inheritance from above. The purpose of object-oriented models is to be more maintainable, flexible and natural (Easterbrook, 2001: 2). The IDEF4 (Integrated Definition Model for Object-Oriented Design) is an example of an object-oriented model – see Figure 2-18 (Borysowich, 2007).

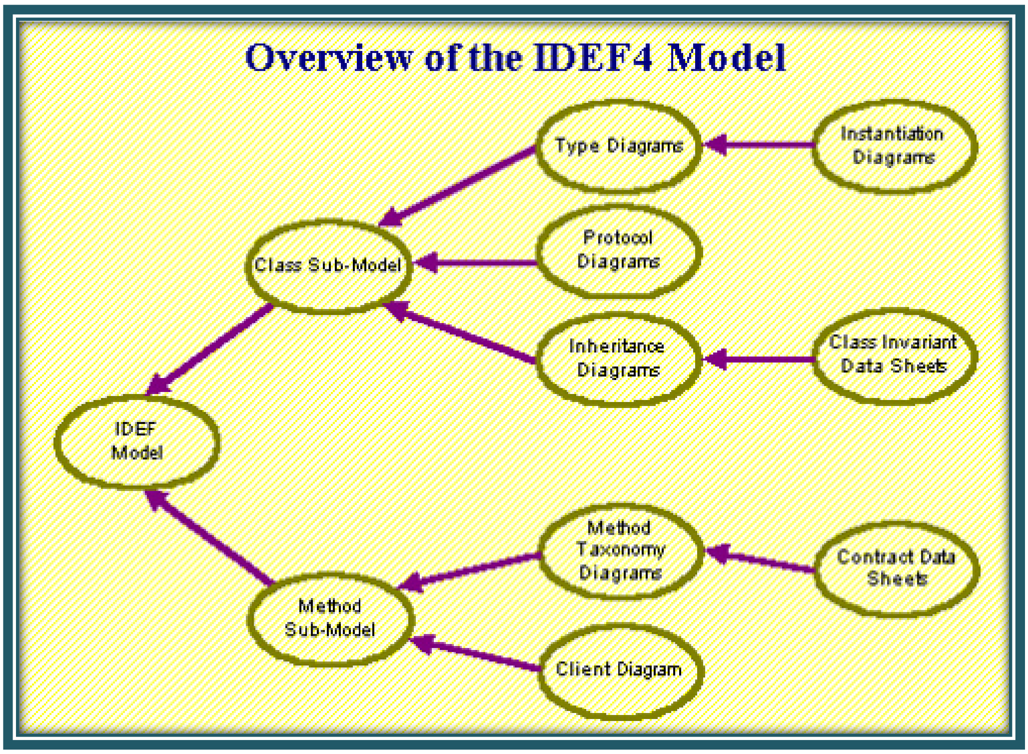


Figure 2-18: Example of Object-Orientated Model (Borysowich, 2007)

These were a few examples of generic models, which could be utilised in enterprise architecture. The last aspect of enterprise architecture to be described in this chapter is the different architectural domains that form part of enterprise architecture.

2.9 LAYERS/DOMAINS OF ENTERPRISE ARCHITECTURE

As part of describing enterprise architecture, the different layers or domains encompassed by enterprise architecture will be discussed from the literature. Different stakeholders observe the enterprise and the architecture from various viewpoints, influenced by their domain of experience (Lankhorst *et al.*, 2005: 53). Bernard (2005: 37) labels these as 'sub-architectures' or 'levels', with distinct functional areas and relationships. O'Rourke (2003: 468) defines a 'domain' as "an area of interest with well-defined boundaries. A domain may contain other domains." The enterprise is the highest level of domain in this context. Lankhorst *et al.* (2005: 87) talk about 'layers', containing internal services, which typically link to other layers by being supported by a layer below or by supporting realisation of a layer above. The term 'architectural domain' will consistently be used throughout this research.

There are quite a number of possible architectural domains which can form part of enterprise architecture. Ross, Weill & Robertson (2006: 48) list four architectural domains: "business process architecture (the activities or tasks composing major business processes identified by the business process owners); data or information architecture (shared data definitions); applications architecture (individual applications and their interfaces); and technology architecture (infrastructure services and the technology standards they are built on)." Sessions (2007: 2) uses the terms, business architectural level, application architectural level and IT architectural level. Gartner (2007: 17) also adds solutions architecture as a cross section of business, information and technology architecture. Jonkers *et al.* (2006: 64) address information, application, process, product and technical architecture as well as their interrelationships.

It is important for an enterprise to capture its understanding of enterprise architecture in a simple one-page core diagram to assist managers and executives to understand their enterprise's EA (Ross, Weill & Robertson, 2006: 50). An initial attempt is depicted in chapter 1, Figure 1-1. Examples of generic depiction of enterprise architecture are shown below in:

- Figure 2-19: Gartner's Representation of Enterprise Architecture (Gartner, 2007: 17) – showing the business context, architecture sphere and solutions architecture;
- Figure 2-20: Sessions's Representation of Enterprise Architecture (Sessions, 2007: 2) – showing enterprise architecture as the overarching architecture;
- Figure 2-21: Jonkers's Representation of Enterprise Architecture (Jonkers *et al.*, 2006: 64) – showing the interrelated architectural domains without any indication of business architecture; and

- Figure 2-22: Representation of Enterprise Architecture from www.e-cio.org (Wu, 2007: 2) – depicts architectures cross-cutting through stovepipe systems and identifies commonalities.

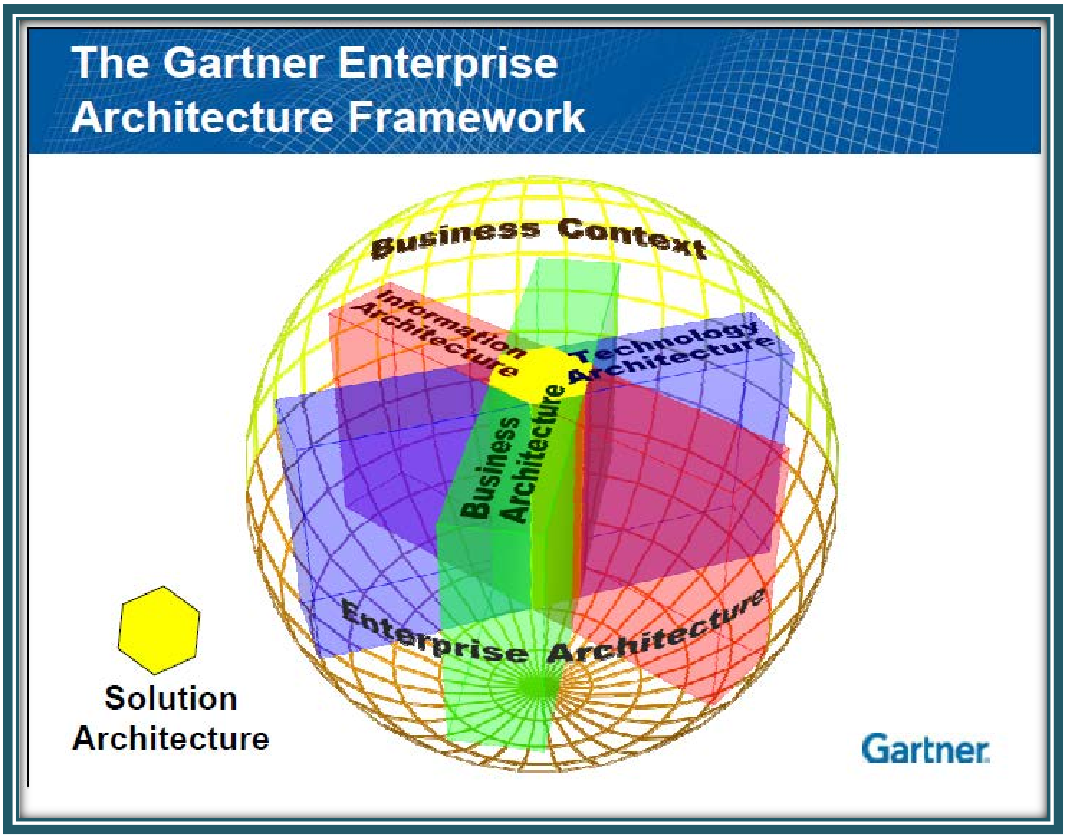


Figure 2-19: Gartner's Representation of Enterprise Architecture (Gartner, 2007: 17)

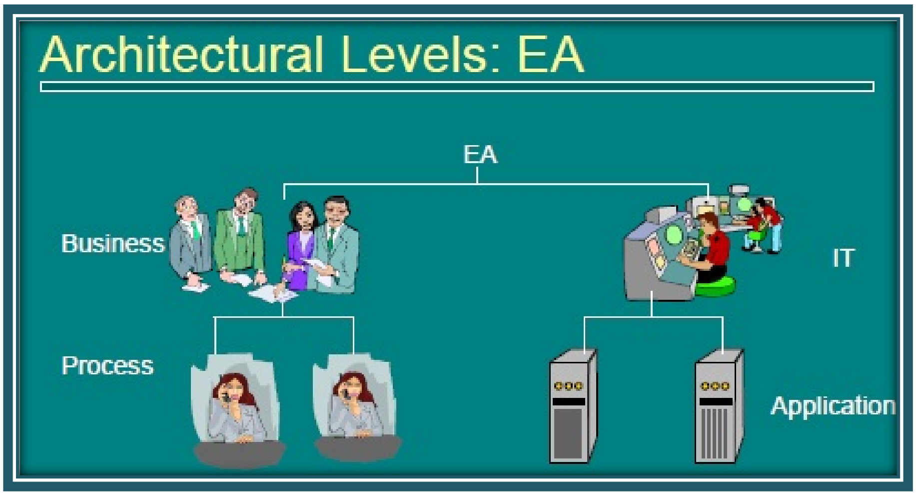


Figure 2-20: Sessions's Representation of Enterprise Architecture (Sessions, 2007: 2)

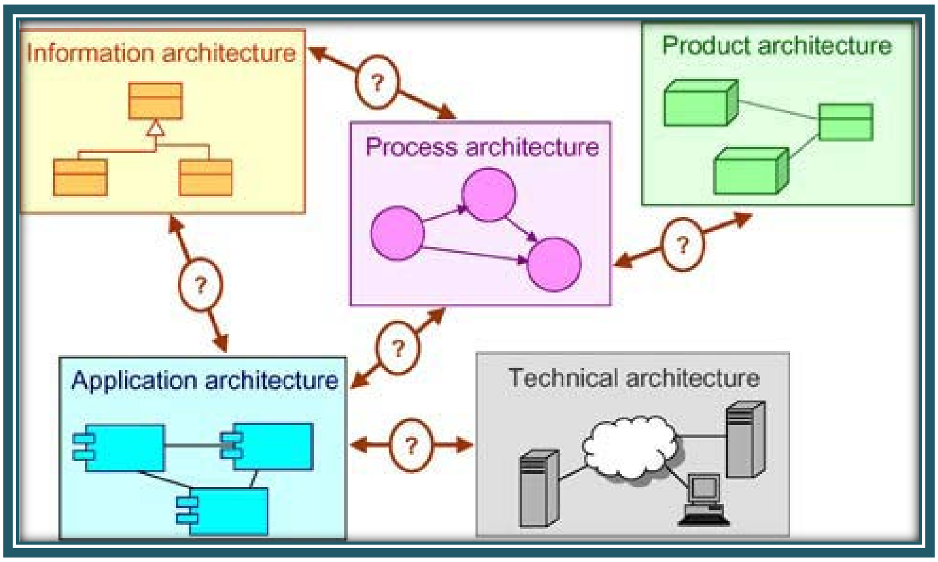


Figure 2-21: Jonkers's Representation of Enterprise Architecture (Jonkers *et al.*, 2006: 64)

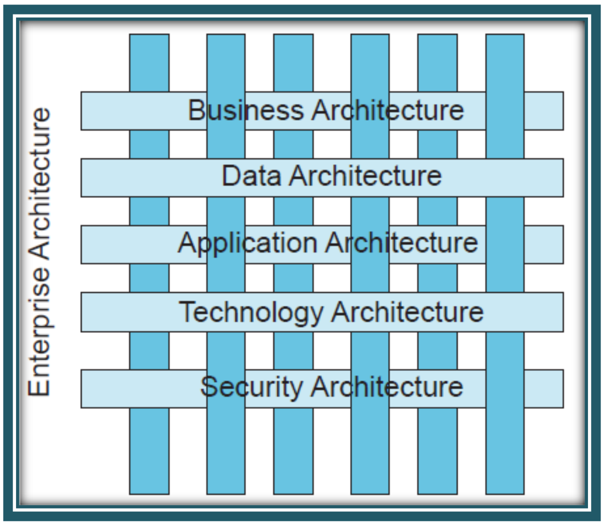


Figure 2-22: Representation of Enterprise Architecture from www.e-cio.org (Wu, 2007: 2)

The next chapters will describe some of these architectural domains, starting with business architecture.

2.10 SUMMARY AND CONCLUSION

2.10.1 Summary

The **first** sub-research question answered in this chapter is: *a) What definitions, frameworks and models are there for enterprise architecture?* This question was answered by exploring a representative sample of the definitions, the history, the existing frameworks and the existing models of enterprise architecture. Furthermore the different domains, which form part of enterprise architecture, were investigated.

- Several **definitions** from literature were cited and analysed. A synthesised definition was compiled and is repeated in Figure 2-23 for ease of reference:

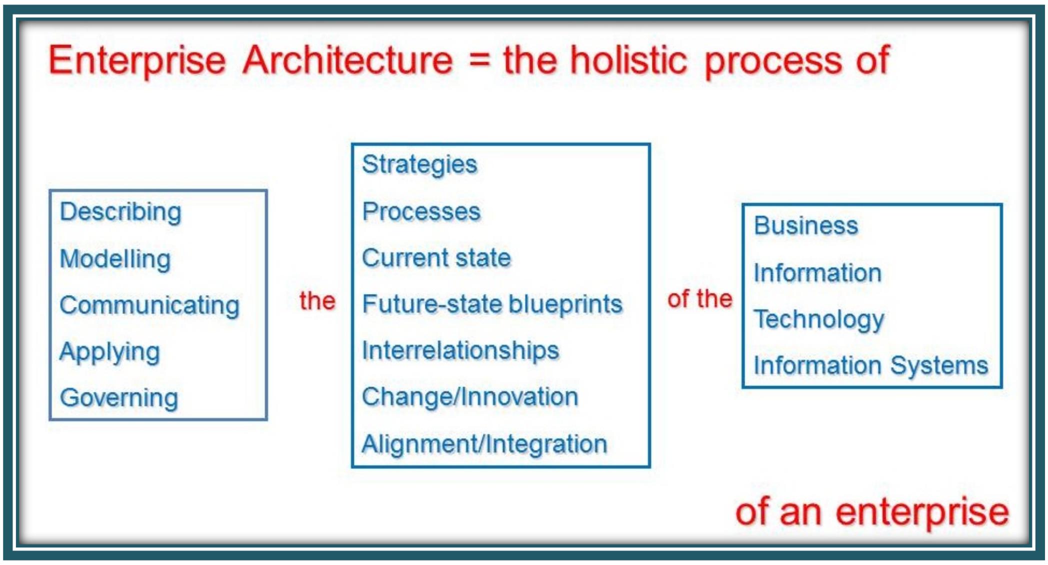


Figure 2-23: Synthesised Definition of Enterprise Architecture

- The **history** was described from 1987 till 2010 in terms of frameworks and their interdependencies. The foreseen future was described based on Gartner's hype cycle.
- The concept of a **framework** was described. The following frameworks were described in terms of overview, scope, views, abstractions, the system development life cycle, strengths and weaknesses: the Zachman Framework, TOGAF, DoDAF and FEAF. A few other frameworks, *i.e.* E2AF, SAGA and GERAM, were briefly discussed.
- The use and interpretation of the term **model** within the enterprise architecture context is not consistent. Generic models can, however, be utilised to describe or design specific artefacts as part of the enterprise architecture process. A short description of the Process Model, the Relational Model, the Causal Loop Diagram Model and the Object-Oriented Model was provided.
- The main architectural **domains** of enterprise architecture are business architecture, information architecture and technology architecture and should be depicted in a high-level core diagram for managers and executives.

It is essential that the role, functions and especially benefits of documenting enterprise architecture are recognised by the enterprise in order for the process to be completed and successful. The **second** sub-research question answered in this chapter is:

b) What are the rationale, purpose and role of enterprise architecture?

The threefold **role** of enterprise architecture within an enterprise is summarised in Table 2-2: Roles of Enterprise Architecture, below:

ROLES	***DESCRIPTION***
Vision	Gaining a holistic view and vision of the future state of the enterprise.
Documented Complexity	Capturing the complexity of the enterprise in a manageable fashion.
Framework	A framework and toolset for implementation and application of enterprise architecture.

Table 2-2: Roles of Enterprise Architecture

The main functions of enterprise architecture have been described. These functions can be converted into multiple benefits for an enterprise by documenting enterprise architecture effectively and efficiently. A summary of the findings is tabled below in Table 2-3: Benefits of Enterprise Architecture.

FUNCTION	DESCRIPTION OF BENEFIT
Alignment	Enabling alignment across all the levels and domains of an enterprise.
Information Provisioning	Provisioning of stable and consistent information on the whole enterprise. Information provisioning for decision support. Providing information and alternatives as a planning tool. Satisfying customer requirements.
Driving Innovation	Insight into and a vehicle for driving innovation and change.
Creating Agility	Enhance responsiveness and agility within the enterprise.
Utilisation of Resources	Improved effective and efficient utilisation of IT resources. Reduction of IT costs. Reduction of duplication.
Process Enhancement	Improved communication processes. Assistance in strategic governance. Optimisation of internal operations and business processes. Consistent and integrated enterprise architecture processes.
Risk Reduction	Increased risk tolerance and insight.
External Linkage	Increased flexibility and knowledge to enhance linking with external partners.

Table 2-3: Benefits of Enterprise Architecture

There are a number of challenges in documenting an enterprise's enterprise architecture. These challenges are summarised below in Table 2-5.

The **third** sub-research question answered in this chapter is: *c) What are the benefits and challenges in documenting enterprise architecture?* Documenting the enterprise's enterprise architecture is a vital component of the architecture process and should focus on the future state, with just the necessary amount of emphasis on the current state. Every function (described in 2.5) will result in a benefit if fulfilled successfully and will not be repeated here. The **benefits** of documenting enterprise architecture are tabled in Table 2-4.

BENEFIT	DESCRIPTION
Execution Foundation	Building a foundation for execution because it maps out processes, data and technology.
Intellectual Capital	Capturing and retaining the existing wealth of intellectual capital in the enterprise.
Value Realisation	A value realisation process, running concurrently with the documenting process, will make the value more visible to the enterprise.
Measurables	Providing metrics to enterprise architecture

Table 2-4: Benefits of Documenting Enterprise Architecture

Documenting an enterprise's enterprise architecture presents its fair share of **challenges**, which are summarised in Table 2-5.

CHALLENGES	DESCRIPTION
Double-barrelled approach	To balance out the increase of IT efficiency while continuing business innovation or in other words complicated systems and complex systems.
Scalability	To create scalable architecture that has the ability to change.
Measuring	To acquire the assessment tools to prove value to enterprise;
Ambiguous documentation	To avoid ambiguous documentation by different stakeholders or sections of the enterprise.
Rate of Change	To deal with the rate of technology and business changes
Maintenance	Maintenance of the models and documentation
Co-operation of enterprise	Commitment and current state of enterprise can impede the architecture process.
Communication	Communication and promotion of enterprise architecture within the enterprise.
Governance	Governance of the enterprise architecture

Table 2-5: Challenges in Documenting Enterprise Architecture

Enterprise architecture has thus been described in terms of definitions, history, role, functions and qualities, documenting, existing frameworks, existing models and the possible architectural domains. Enterprise architecture consists of several components or domains. The first of these is business architecture and will be discussed in more detail in the following chapter.

2.10.2 <u>Conclusion</u>

Three sub-research questions, pertaining to the enterprise architecture environment, were addressed in this chapter. This provided a part of the background and context necessary to enable the goal of designing and testing an architecture framework, which will address especially the integration or linkages between the different architectural domains. Background on the definition and role of enterprise architecture, the documentation thereof and an understanding of existing frameworks and models are essential in order to create a feasible new or modified framework.

SECTION A: LITERATURE STUDY

This section researches the current status of enterprise architecture. The research includes enterprise architecture, business architecture, information architecture and technology architecture. This encompasses the history, roles, definitions, documenting, frameworks, models, ontologies and descriptive languages. The outcomes for each architectural domain are:

- syntheses of defining terms used in literature;
- proposed, new, synthesized definition;
- overview of existing frameworks, models, ontologies and languages;

Furthermore, the relationships between the architectural domains are researched in terms of depictions, integration strategies and interlinking mechanisms.

This section is divided into five chapters.

The next section will address the *Design of a Relational Metamodel* to provide a possible solution for documenting the relations between the different architectural domains.

3 BUSINESS ARCHITECTURE

3.1 INTRODUCTION

Enterprise architecture was discussed in detail in the previous chapter. The focus now shifts to providing a high-level overview of some of the main architectural domains. Business architecture is one of the architectural domains of enterprise architecture and the subject of the next sub-research question:

d) What definitions, benefits and models are there for business architecture?

Development of the business architecture of an enterprise is a step necessary to define the holistic future state of the enterprise. It provides "the business view of the future state by defining the business models and process flows that change across the enterprise to achieve the organization's strategic goals" and is a prerequisite for developing a successful enterprise architecture (Gartner, 2008a: 2).

The current status of business architecture is hereby described by means of a critical analysis of existing literature, through exploring the definitions, role and benefits of business architecture. Some of the existing frameworks and models are also described briefly. This will provide essential background to create an understanding of the environment and will be used as input in creating a model to facilitate the integration between the different architectural layers or domains.

3.2 DEFINITIONS OF BUSINESS ARCHITECTURE

The existing definitions in the literature will be explored in order to be able to compile a working definition of business architecture. The terms '*business architecture'* (BA), '*business enterprise architecture'* and '*enterprise business architecture'* (EBA) are used interchangeably by authors in this domain (Handler, 2001; Kang, Lee & Kim, 2010; Versteeg & Bouwman, 2006).

3.2.1 Definitions in the Literature

Although the term '*business architecture'* appears in existing publications, it is not unambiguously defined (Orr, 2008: 4). There are differences between approaches (physical, logical, strategic alignment), degree of specification (conceptual, modelling, strategy, processes), the layers (business, information, technology) and the breadth of view (own enterprise, whole supply chain) (Versteeg & Bouwman, 2006: 92). The following definitions of business architecture were chosen from the literature and are discussed chronologically below.

Handler from the Meta Group (2001): "An enterprise business architecture (EBA) is the expression of the enterprise's key business strategies and their impact on business functions and processes:

- It typically consists of the current and future-state models of business functions, processes, and information value chains.
- It is implemented through the enterprise, enterprise information architecture, enterprisewide technical architecture, and application portfolio.
- It defines the business design for sustainable competitive advantage".

This definition presents three problems: cumbersome wording, the omission of mentioning resources and a lack of addressing business architecture as a process.

Whittle & Myrick (2005: 31) use the term 'enterprise business architecture' as defining "the enterprise value streams and their relationships to all external entities and other enterprise value streams and the events that trigger instantiation. It is a definition of what the enterprise must produce to satisfy its customers, compete in a market, deal with its suppliers, sustain operations, and care for its employees. It is composed of models of architectures, workflows, and events." They see EBA as the functional architecture which has links from the corporate strategy and business environment through to the process initiatives, including software development domains. Again, this definition omits the process nature of business architecture and direct reference to resources. In the definition itself no reference is made to strategy or objectives.

Versteeg & Bouwman (2006: 93) describe business architecture as "the grouping of business functions and related business objects into clusters (business domains) over which meaningful accountability can be taken as depicted in the high level description of the related business processes”. This definition focuses on business processes and neglects the strategic objectives, managing of resources and the links between entities.

Dols (2008: 31) summarises by stating that business architecture is “designing an organization and its processes to realize its business vision and objectives.” He further refers to the following components: vision and objectives, strategies, capabilities, processes, and information. Although the definition is very concise, it touches on quite a few of the important aspects except for the managing of resources and the relationships between entities.

Gartner defines EBA as "that part of the enterprise architecture (EA) process that describes – through a set of requirements, principles and models – the future state, current state and guidance necessary to flexibly evolve and optimize business dimensions (people, processes, financial resources and the organization) to achieve effective enterprise change" (Blechar, 2009: 3). They differentiate between four main dimensions, *i.e.* people, financials, organisations and processes. Although interlinking is not specifically mentioned in this otherwise comprehensive definition, it is incorporated in the organisation level and cross-cutting functions as in Figure 3-1.

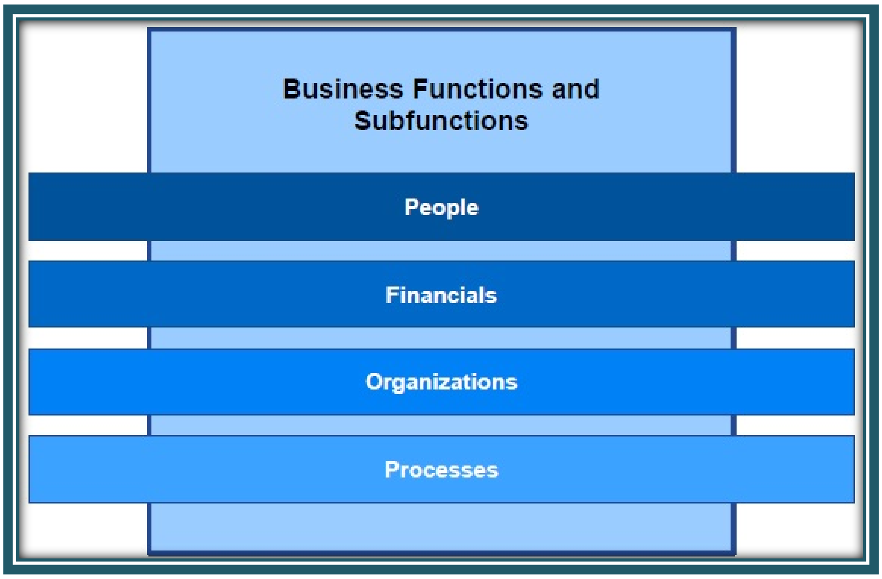


Figure 3-1: Gartner's Dimensions of EBA (Blechar, 2009: 4)

“Graphical representation of a business model, showing the networks through which authority, information, and work flows [sic] in a firm. It serves as the blueprint of a firm's business structure, and clarifies how the firm's activities and policies will affect its defined objectives" (Business Dictionary, 2010). Although the definition addresses most of the aspects, business architecture is much more than just a graphical representation.

Kang, Lee & Kim (2010: 3275) define business architecture as "the EA to manage strategies, processes, and resources systematically, and to maintain relationships among them." They also provide a high-level diagram of their view of business enterprise architecture, consisting of business, IT and human resource architecture as in Figure 3-2. This is a short and to the point definition.

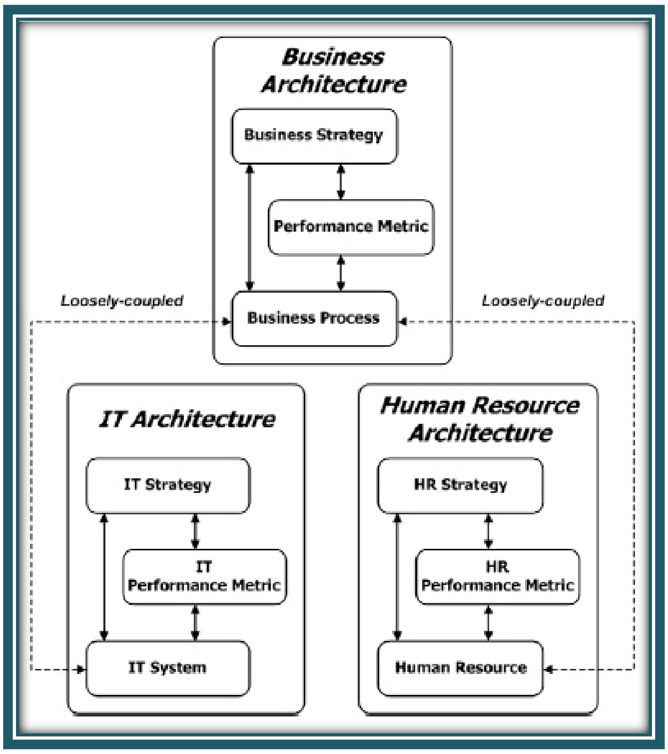


Figure 3-2: EA containing Business Architecture (Kang, Lee & Kim, 2010: 3275)

The content of the above definitions was analysed based on the terminology used. The matrix in Table 3-1 provides a summary of terms, synonyms and verbs used. The terms were grouped together under output formats, actions, objects and scope. The group objects were subdivided into strategy and business components. The terms most frequently used are: business processes, followed by interrelations and then business strategies and models. The category 'resources' includes people, and financial resources.

Authors			Handler	Whittle & Myrick	Versteeg & Bouwman	Dols	Gartner	Business Dictionary	Kang, Lee & Kim
Output formats		Models	•				•	•	
		Produce		•					
		Business design	•			•			
		Descriptions			•		•		
		Graphical representation						•	
		Blueprint						•	
		Requirements					•		
		Principles					•		
		Expression	•						
Actions		Impacts	•					•	
		Implements	•						
		Defines	•	•					
		Organises			•				
		Manages							•
		Changes					•		
Objects	Plan/ Strategy	Business strategies	•	•		•			•
		Vision				•			
		Objectives				•		•	
		Competitive advantage	•						
		Policies						•	
	Business	Value chains	•	•					
		External entities		•					
		Business functions	•		•				
		Business processes	•	•	•	•	•		•
		Workflow		•				•	
		Events		•					
		Business objects			•				
		Information	•			•		•	
		Business capabilities				•			
		Organisation structure					•	•	
		Resources		o			•		•
Scope		Current state	•				•		
		Future state	•				•		
		Enterprisewide	•						
		Relationships	•	o	•			•	•
		Process approach					•		
	o	Implied in definition	•	Mentioned in definition					

Table 3-1: Analysis of Business Architecture Definitions

3.2.2 <u>Synthesised Definition</u>

Based on a synthesis of the literature definitions above, a working definition for business architecture was compiled and is illustrated in Figure 3-3.

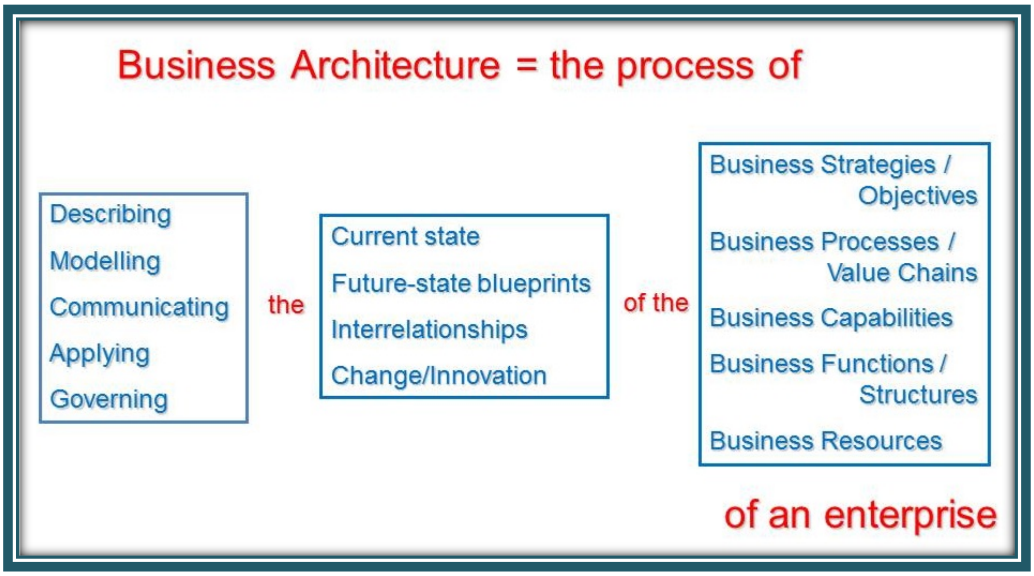


Figure 3-3: Synthesised Definition of Business Architecture

The proposed definition in paragraph format is:

> *Business architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships and change/innovation of the business strategies/objectives, processes/value chains, capabilities, functions/structure and resources (human & finance) of an enterprise.*

3.3 THE ROLE AND BENEFITS OF BUSINESS ARCHITECTURE

In order to gain the most benefit from business architecture, an enterprise should refrain from making use of an ad hoc approach. It is easy for an enterprise to fall into the pitfall of giving attention to business architecture on a per-project basis or in certain lines of business. This could result in resources "being wasted because there is no coordination of business architecture efforts that will leverage those efforts" (Handler, 2001).

Business architecture has a specific role to play within enterprise architecture. Some of the facets of this **role** are:

- Provisioning assistance to enterprises in coping with radically changing business environments by working out corresponding strategies;
- creating flexible processes; and
- organising resources and maintaining of the relations (Kang, Lee & Kim, 2010: 3275).

- Creating business artefacts, like the descriptions of services, the definitions of the goals, the specifications of the processes and the organisational structure (Aier & Winter, 2009: 151).

- Clarifying complexity within an enterprise;
- gaining insight into the business strategies;
- provisioning a "starting point from which to develop subsequent functional, information, process and application architectures"; and
- enabling enterprises to react swiftly to changes in the environment (Versteeg & Bouwman, 2006: 91).

- Providing a mechanism of predicting corporate outcomes;
- lessening the overwhelmed and chaotic state of enterprises; and
- creating causality, connectivity and understanding of the internal and external relationships (Whittle & Myrick, 2005: 3).

- Outlining "a new *work environment* with operational principles, processes and tools" (University of California, 2000: 3).

- Generating descriptions and definitions of the business; and
- providing assistance with acquisitions, mergers and consolidation of enterprises (Lloyd, 2006: 4,15).

Some of the **benefits** of business architecture are:

- It helps to structure the roles and responsibilities within an enterprise;
- it provides guidance and shape to possible areas of outsourcing; and

- it contributes to the governance of ICT (Information and Communications Technologies) "in order to orchestrate the resources for critical business activities" (Versteeg & Bouwman, 2006: 91).

- It creates a context for:
  - setting priorities,
  - managing growth,
  - controlling costs, and
  - improving the work environment; and
- it provides an opportunity to implement best business practices (University of California, 2000: iii).

- It creates a value proposition consisting of capabilities and relevant metrics; and
- it generates a capability-based operating model (Lloyd, 2006: 13).

- It brings about solution delivery cost reduction; and
- it decreases solution delivery cycle times (Gartner, 2008a: 17; Handler, 2001).

- "It ensures that changes and enhancements to business process and organization are fully optimized with the information and technology direction in support of the business strategy" (Gartner, 2008a: 3).

Business architecture thus has an important role within an enterprise to provide flexibility in change, to organise, to create business artefacts and to gain insight and understanding of the business. Business architecture can produce benefits mainly in assisting in structuring, guiding, shaping, managing and improving the enterprise.

3.4 EXISTING FRAMEWORKS AND MODELS

There are existing frameworks and models to describe the artefacts of business architecture. Artefacts can vary depending on the framework or model used. Examples of artefact types are catalogues, matrices, core and extension diagrams and examples of artefacts are a business service catalogue, a stakeholder map matrix, a data dissemination diagram and a product life cycle diagram (The Open Group, 2009b).

In order to explore the existing frameworks and models of business architecture, different levels of abstraction need to be addressed:

- First, a framework is a skeletal structure to facilitate describing the architecture. The business architecture component of the enterprise architecture frameworks, where applicable, will be discussed first, followed by other business frameworks.
- Second, a few models, which can be used for providing content to the framework, are touched upon.
- Third, ontologies will be discussed to express the relationships between elements.
- Fourth, some descriptive languages, which can be used to express the content of the frameworks or models, are listed.

3.4.1 Business Components of Enterprise Architecture Frameworks

Some of the enterprise architecture frameworks discussed in paragraph 2.7 have specific business architecture components and will be discussed briefly here.

3.4.1.1 Zachman Framework

The Zachman Framework has a row dedicated to the business concepts from an owner or executive perspective. This view is composed of:

- business resource/inventory definition – addressing the 'what question' by relating the business entities through the business relationships to other peer business entities;
- business process definition – addressing the 'how question' by relating business transformations through business inputs to other peer business transformations;
- business network definition – addressing the 'where question' by relating business locations through business connections to other peer business locations;
- business organisation/work flow definition – addressing the 'who question' by relating business roles through business work to other peer business roles;
- business cycle/timing definition – addressing the 'when question' by relating business cycles through business moments to other peer business cycles.; and

- business motivation/strategy definition – addressing the 'why question' by relating business ends through business means to other peer business ends (Anon., 2008b; Hokel, 2010: 4; Locke, 2010).

3.4.1.2 TOGAF – Business Architecture

Business architecture is phase B of TOGAF's Architecture Development Method (ADM), following after architecture vision. The objectives are to describe the baseline business architecture, to develop the target business architecture, to analyse the gaps between the baseline and the target, to develop the relevant stakeholder viewpoints and to select relevant tools and techniques. TOGAF views business architecture as a prerequisite for the other architectural domains (The Open Group, 2009c: 94).

A variety of modelling tools can be employed to do the modelling (The Open Group, 2009c: 96):

- Activity Models – to describe the functions and data exchange of the enterprise's business activities in a hierarchical manner.
- Use-Case Models – to describe the business processes of an enterprise in terms of use-cases and actors corresponding to business processes and the enterprise.
- Class Models – to describe the static information and relationships between information as well as informational behaviours.

3.4.1.3 DoDAF

DoDAF does not contain a section specifically called 'business architecture', but the 'Operational View' contains a list of useful artefacts, for example:

- High-Level Operational Concept Graphic,
- Operational Node Connectivity,
- Organisational Relationships Chart,
- Operational Activity Model,
- Operational Rules Model,
- Operational State Transition Description, and
- Operational Event-Trace Description (USA. Department of Defence, 2007b: 1-10).

A variety of modelling tools can be employed to do the modelling (USA. Department of Defence, 2007b: 2-4):

- Structured Models – to provide a process driven hierarchical decomposition of functional processes,
- Object-Orientated Architectures – to describe the operational need in the context of its use, based on the concepts of data abstraction,
- Activity-Based Methodology (ABM) – to enable development and analysis of integrated architectures using a disciplined approach,
- Architecture Specification Model (ASM) – to provide a common set of semantics.

3.4.1.4 FEAF

FEAF makes use of the Business Reference Model (BRM) for creating the business architecture section of enterprise architecture. The BRM is a "Function-driven framework for describing business operations of the Federal government independent of the agencies that perform them" (USA. FEA Program Management Office, 2004: 8). Figure 3-4 depicts the different components from the management, support, mode and descriptions of services.

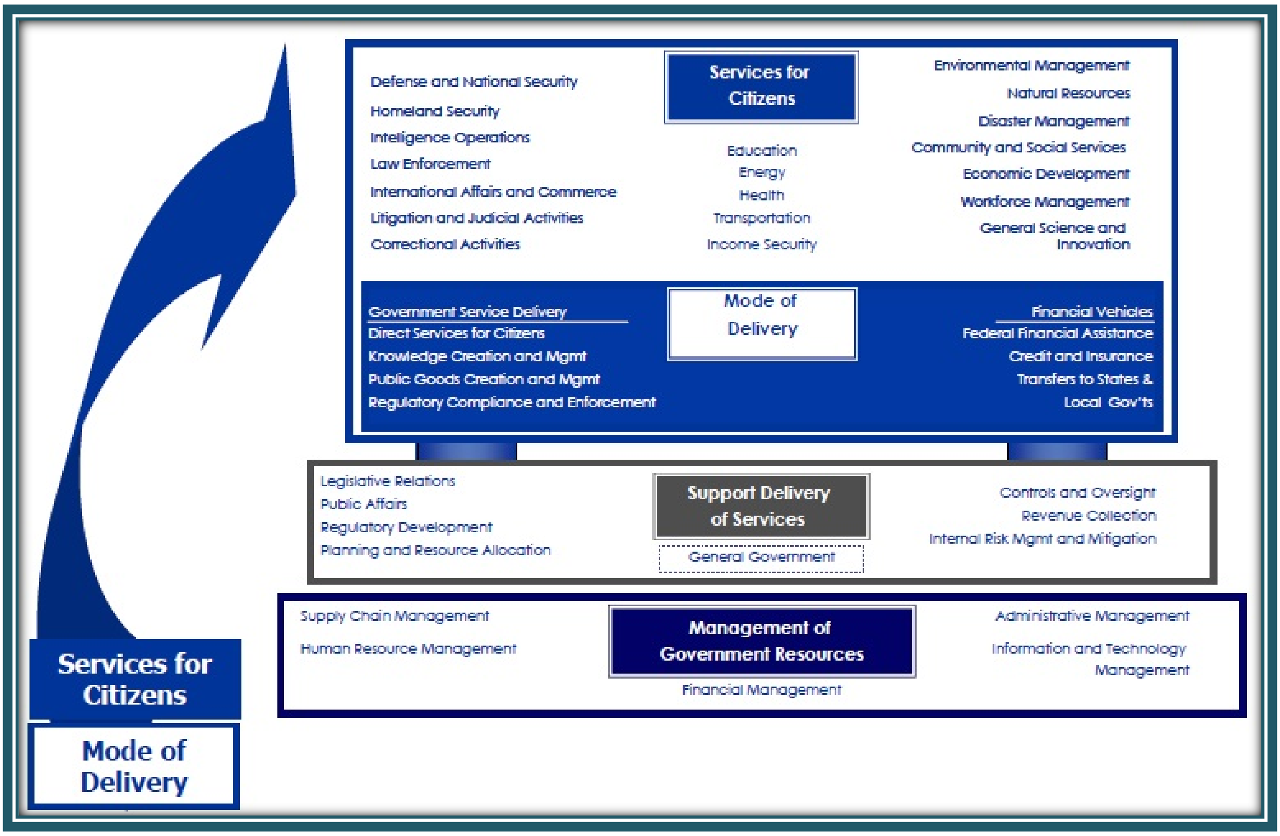


Figure 3-4: FEAF's BRM (USA. FEA Program Management Office, 2004: 10)

3.4.1.5 Gartner Business Architecture Framework

The Gartner way of approaching the business architecture section of enterprise architecture is called the 'Business Viewpoint': "Business viewpoint is focused on identifying, describing and optimizing critical business process topology." The outputs or artefacts of the business viewpoint journey are business requirements, principles (standards and guidelines) and models (blueprints and models) (Gartner, 2008a: 4).

It makes use of different perspectives as in Figure 3-5 which are aligned with the current strategic direction of the enterprise:

- People: the roles, preferences, competencies, behaviours, preferences and networking of the people;
- Organisation: structures (formal and informal), roles and responsibilities, governance and operating models of the enterprise;
- Business Process: the logical relationship between processes and the intersection and friction points between processes, modelled by the Business Anchor Model

(BAM) (see bottom left of Figure 3-5). The BAM documents the core value chain business functions and their supporting services, which can include entities outside the business (Gartner, 2008a: 3).

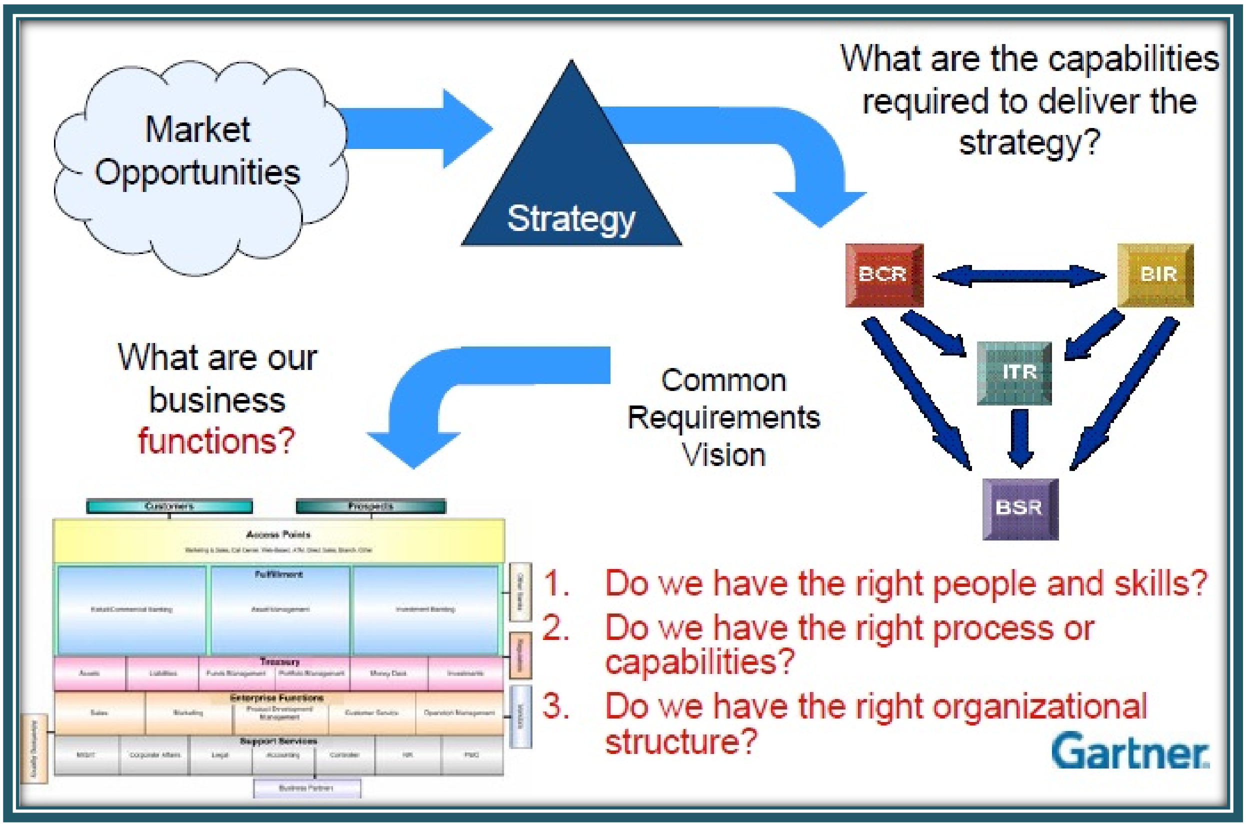


Figure 3-5: Business Viewpoint (Gartner, 2008: 5)

The business viewpoint has different layers of abstraction (Gartner, 2008a: 10):

- The highest level of abstraction of this viewpoint is the **business context**. The business context should provide the foundational assumptions for the entire future-state architecture. Here the strategic requirements of the enterprise are analysed and relevant architecture principles are set while taking environmental trends into account. The CRV (Common Requirements Vision) is a defined process for deriving architecture principles from the business strategy. Furthermore the business functions, which are necessary to fulfil the business strategy should be analysed including the high-level requirements of each business function.
- The next layer of abstraction is the **conceptual layer**, which identifies the critical business processes, describes the process topology and flow as well as the interrelationships between processes and functions and the link to strategy (Gartner, 2008a: 13).

- The **logical layer** includes business process patterns indicating process flows across business functions and can be illustrated by using a process pattern diagram (Gartner, 2008a: 14).
- The **implementation layer** can make use of flow diagrams or flow maps to plot specific processes and contribute value to the business perspective by indicating the relevant business functions (Gartner, 2008a: 15).
- All levels of abstraction contain **artefacts**, some of which are derived from higher levels. These artefacts should be traceable from the conceptual design layer to the implementation layer, which links back to the business context.

Gartner (2008a: 16) views the scope of business architecture as ending with "the creation of optimized models of business process, duly noting dependencies, and in the context of the business strategy". Business process management begins with these optimised models and ends with automated processes.

3.4.2 Business Architecture Frameworks

There are different ways of approaching business architecture. McWhorter (2008: 11) has a useful perspective to view business architecture by distinguishing between motivational, operational and analytical aspects, see Figure 3-6. The motivational aspect includes items like vision statements, goals, strategies and tasks. The operational aspect includes items like workflow, rules, task enablement and specifications. Lastly the analytical aspect includes items like value chains, root-cause analysis and market segment analysis.

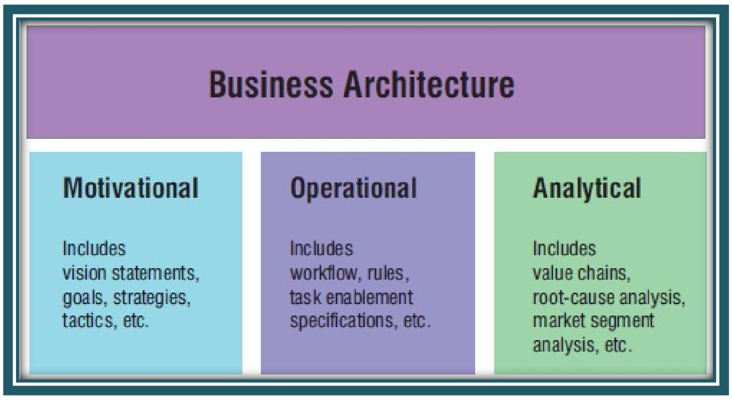


Figure 3-6: Aspects of Business Architecture (McWhorter, 2008: 11)

Several possible frameworks will be briefly described here, including among others the New Business Architecture for the University of California (UC) and the Agile Business Process Modelling Framework.

3.4.2.1 Microsoft Motion Business Architecture Methodology

Microsoft Motion is a business architecture methodology developed by Microsoft to describe the architecture of the business domain of an enterprise. The basic module map is illustrated in Figure 3-7. For example: to fulfil a demand (module 3), the enterprise's business capabilities should be mapped out in terms of people, process and platform (Lloyd, 2006).

Figure 3-7: Microsoft Motion – High Level Generic Module Map (Lloyd, 2006: 7)

A lighter methodology, Motion Lite, is also available. This focuses on capturing a high-level capability map and then on the business value, maturity, and interconnectedness of these capabilities. As an example, one block in Figure 3-7 can be used and applied to the different levels below it, to create a capability map:

- start with block *3. Fulfil Demand* (level 1) and map out the level 2 capabilities;
- then choose one of these level 2 capabilities, for example – *Procurement*, and map out the level 3 capabilities;

- then choose one of these level 3 capabilities, for example – *Purchase Resources*, and map out the level 4 capabilities;
- then choose one of these level 4 capabilities, for example – *Manage Suppliers* and map out level 5 capabilities; and
- finally use this map to identify business values and improvement projects (Merrifield & Tobey, 2006).

According to McGowan (2007) Microsoft Motion now forms part of Microsoft Services Business Architecture (MSBA).

3.4.2.2 A New Business Architecture for UC

The University of California followed a lengthy process to develop what they call the 'New Business Architecture'. The framework is depicted in Figure 3-8. The central section is the 'Business Portal', which should be the gateway to the support operations as it integrates the transactional systems with policies, training and tools. The other five components can be very briefly summarised as:

- People: establish the enterprise "as a competitive employer and provide staff the tools to succeed";
- Processes & Policies: "Redesign processes to facilitate getting work done rather than cover all potential transactions";
- Enabling Technology: Provide effective tools while ensuring a secure environment;
- Financial Systems & Reporting: "Employ emerging standards to improve integration of financial data"; and
- Organisational Performance & Controls: "a component that measures how well we are able to achieve our business objectives, while strengthening our financial controls" (University of California, 2000).

Figure 3-8: The New Business Architecture for UC (University of California, 2000: 7)

3.4.2.3 Agile Business Process Modelling Framework

This approach advocates the use of agile modelling combined with a business modelling framework to put just enough discipline in the 'freedom' of agile modelling (Anon., 2009). "Agile Modeling (AM) is a practice-based methodology for effective modeling and documentation of software-based systems. At a high level AM is a collection of best practises. At a more detail level AM is a collection of values, principles, and practices for modeling software that can be applied on a software development project in an effective and light-weight manner" (Ambler, 2011).

The agile business process modelling framework is illustrated in Figure 3-9. This framework guides the populating thereof through agile modelling. It proposes three levels of business modelling:

- Organisational Level: the highest level addressing interactions between units and the interactions with external agents. It also addresses the fundamental functions and variables.
- Process Level: the middle level addresses the workflow models and process maps in supporting the goals of the enterprise. Processes are described in terms of cross-functionality, inputs, outcomes, customers, measurable and triggers.

- Job Level: the lowest level describes the activities for each job and the relevant responsibilities for the goals to be achieved (Anon., 2009).

Domain / Level	Strategy	Work Flow	Information	Management
Organisation	Organisation Strategy and Goals	Organisation Structure and Flow	Business Objects Model	Organisation Management
Process	Process Goals and Problems	Process Flows and Structure	Conceptual Data Model and Flows	Process Management
Performer	Job Goals and Problems	Activity Work Flow	Logical Data Model and Flows	Job Management

Figure 3-9: Agile Business Modelling Framework (redrawn from Anon, 2009)

The framework also contains four views on the abovementioned levels:

- Strategy: approaches the levels from the view of a business strategy, goals, problems and critical success factors.
- Work Flow: approaches the levels from the perspective of an organisational structure, process and activities work flow as well as business rules.
- Information: attends to the perspective of information and materials, model and flows, and the conceptual and logical data model and flow.
- Management: addresses enterprise management, process management, job management and information management (Anon., 2009).

3.4.2.4 Business Motivation Model (BMM)

The Business Motivation Model (BMM) is published by the Business Rules Group and the Object Management Group (OMG). It provides an underlying architectural metamodel for describing the business architecture of an enterprise. There are two major areas of the BMM, namely: the ends and means of business plans and the influencers that shape the elements of the business plans (Rosen, 2008: 6).

The main concepts of BMM are illustrated in Figure 3-10 and are summarised below:

- Ends: *what* the enterprise desires to accomplish (it can be a device, capability, technique, agency or method); encompassing:
  - vision: overall image about what the enterprise wants to be (amplified by goals and operationalised by missions);
  - goal: statement about a state/condition of the enterprise to be brought about through appropriate means; and
  - objective: statement of a measurable, attainable, time-specific target the enterprise seeks to achieve to accomplish its goals.
- Means:
  - mission: the enterprise's plans for achieving the vision;
  - strategy: essential courses of action or approaches to achieve the ends or goals;
  - tactic: details and implementation of the strategies;
  - policy: non-actionable directive to govern strategies and tactics; and
  - business rule: directive to influence/guide business behaviour in support of policies (Rosen, 2008: 7-8).

"The Business Motivation Model provides an industry standard mechanism for rigorous and formal analysis and deserves inclusion in the business architect's toolkit" (Rosen, 2008: 10).

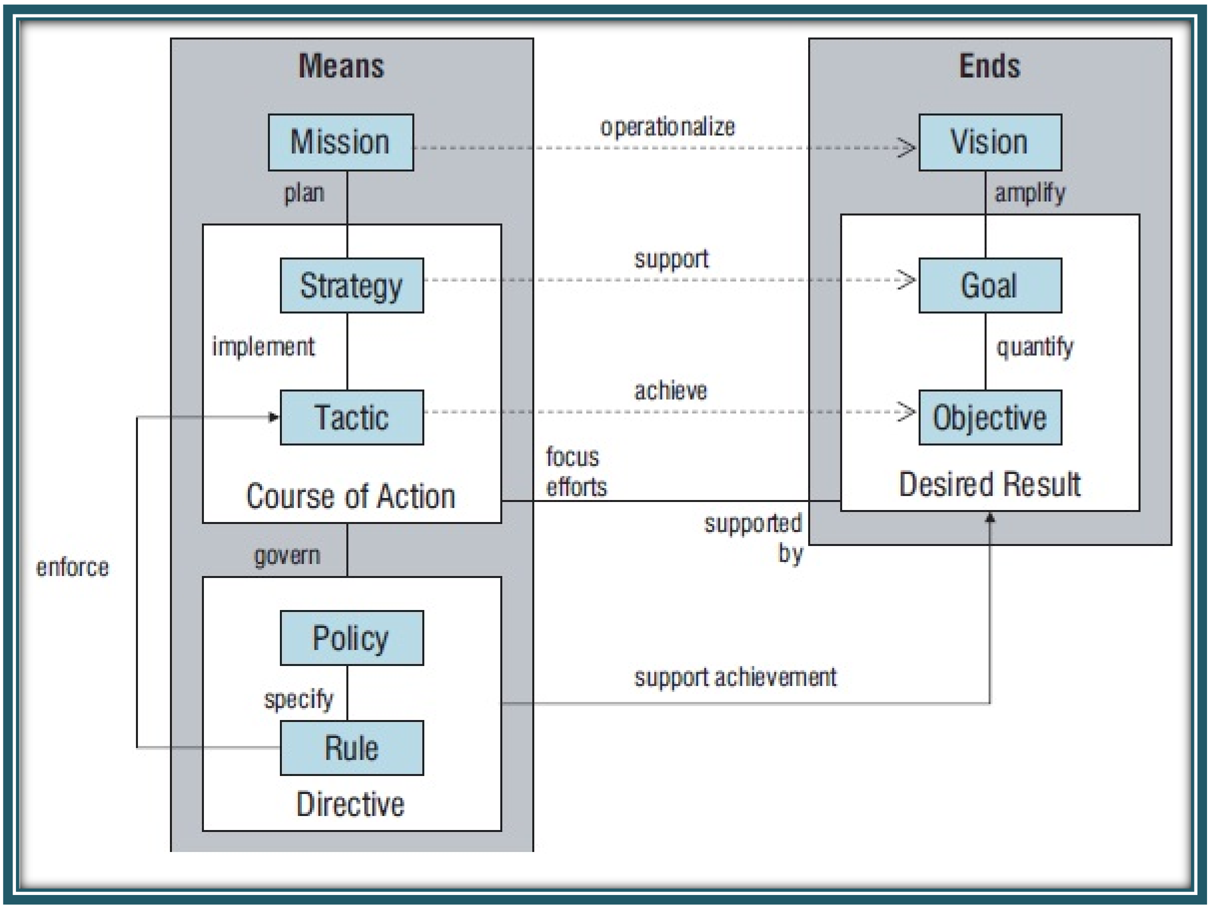


Figure 3-10: The Concepts of BMM (Rosen, 2008: 7)

3.4.3 <u>Modelling Languages</u>

Models, as reasoned in 2.8, are used to fill in the different structures and sections of a framework. The generic models, as discussed in 2.8, can also be used for business architecture and are summarised as follows:

- <u>Process Model</u> – modelling the processes in an enterprise while being independent of the organisational structure. Overlays for organisational structure, projects (requests and actual) or supported applications can be superimposed on the process model (Bedwell, 2006: 17-21).
- <u>Relational Model</u> – a structured way of documenting the pieces of information in relation towards each other, which supports simple, powerful querying of data (Ramakrishnan, s.a.: 14).
- <u>Causal Loop Diagram</u> – depicts the cause and effect of relationships between systems of related variables in terms of directions and time delays, and will show simultaneous causal relationships (Vail, 2002: 10).

- Object-Oriented Model – contains objects (nearly anything can be an object) within objects to an arbitrarily deep level of nesting, organised into classes, where each object has its own unique identity, independent of the values it contains (Zaïane, 1995).

There are other models that can also be used to fill in the sections of a framework. One of these is Pi-Calculus Process Algebra. According to Parrow (2001: 480) "π-calculus is a process algebra where processes interact by sending communication links to each other" and a "mathematical model of processes whose interconnections change as they interact". The basic computational step is the transfer between two processes of a communication link, which can be reused with other processes/parties.

Another model is the Object Management Group's Business Process Model and Notation (BPMN). BPMN "will provide businesses with the capability of understanding their business procedures in a graphical notation and will give organizations the ability to communicate these procedures in a standard manner." Furthermore the "graphical notation will facilitate the understanding of the performance collaborations and business transactions between the organizations" (Object Management Group, 2008a).

One of the latest models is the Capability Model, first being alluded to by Microsoft's Motion. A capability is "the quality of being capable" and "a capacity for being used or developed" (Webster's New World Dictionary, 2011). According to Keller (2009) a capability consists of a verb; covers people, processes and support; and can include performance metrics, service levels and compliance criteria. Gartner (Burton, 2012: 5) describes a verb-noun expression of business capabilities, for example, service customers or manage brand. Homann & Tobey (2006) argue that a business architecture model, making use of a network of capabilities, provides a foundation that is ideally aligned with service-orientation, because they share critical attributes such as external, observable, and measurable behaviour. "Depending on the hierarchy levels you use for the decomposition of your capabilities you end up with catalogs (or call them trees) of something up to 2.000 and more capabilities – grouped hierarchically from the domains down to maybe 5-7 levels" (Keller, 2009). Capabilities could be used to produce a so-called heat map of the capability landscape of an enterprise, see example in Figure 3-11. The red colour could, for example, indicate high maintenance cost and low customer satisfaction.

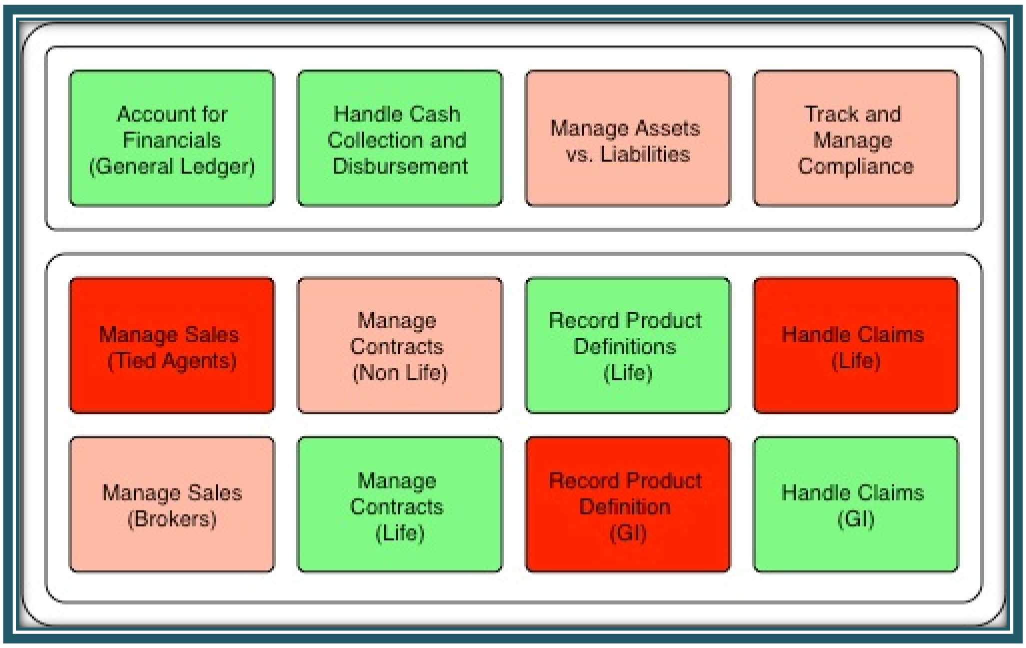


Figure 3-11: Capability Heat Map (Keller, 2009)

There is a variety of descriptive languages, which can be used to express the content of the frameworks or models. These languages could potentially be used to supply a link between the business architecture and the other architectural layers. As they are part of the implementation a few of the existing descriptive languages are only listed below:

- Business Process Execution Language (BPEL) – deals with the functional aspects of business processes (such as flow control, conversations, correlation, faults) and “provides an ‘orchestration engine’ for describing exchanges of information internally or externally” (Cobban, 2004).
- Semantics of Business Vocabulary and Business Rules (SBVR) – “defines the vocabulary and rules for documenting the semantics of business vocabularies, business facts, and business rules; as well as an XMI (XML Metadata Interchange) schema for the interchange of business vocabularies and business rules among organizations and between software tools” (Object Management Group, 2008b: 3).
- Archimate – is a metamodel and “can show a top-down view of the actors, roles, and business services and functions, as well as how they are expected to be supported by new and already existing application and infrastructure services, and how they interface with other application services” (Ricca, 2011).

3.4.4 Ontologies

Ontologies are used to provide relationships between elements and semantics. Ontology is "the branch of metaphysics dealing with the nature of being, reality, or ultimate substance" (Webster's New World Dictionary, 2011) or "Study of the nature of being or existence, or the assumptions underlying a classification scheme, concept, system, or theory" (Business Dictionary, 2010). Different approaches to ontology includes the philosophical, systemic and enterprise ontology (Dietz *et al.*, 2013). Ontology uses the origins of a concept to classify the concept into a schema and to portray the interrelations.

3.4.4.1 Fact-Based Ontologies

Kang, Lee & Kim (2010: 3275) describe a metamodel, which contains:

- architectural model describing the enterprise elements;
- fact-based ontologies to express the relationships between elements and their semantics (described in more detail below); and
- a matrix-shaped alignment table to provide an overview of the relationships at a glance (for example, mapping the strategies onto the processes).

This metamodel addresses "business strategies, performance metrics, business processes, and their relationships" as depicted in Figure 3-12.

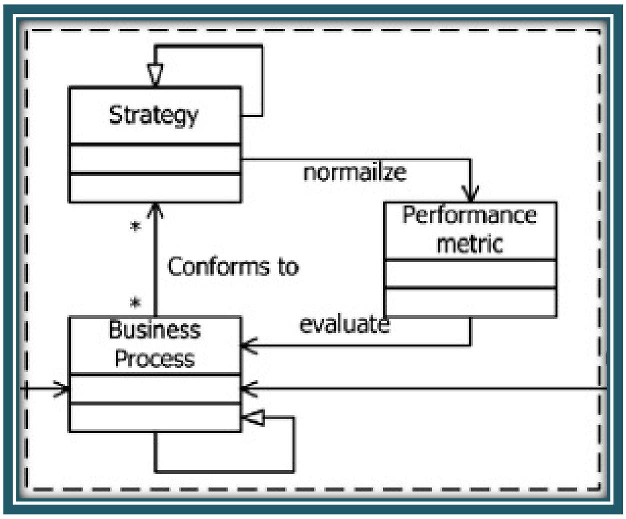
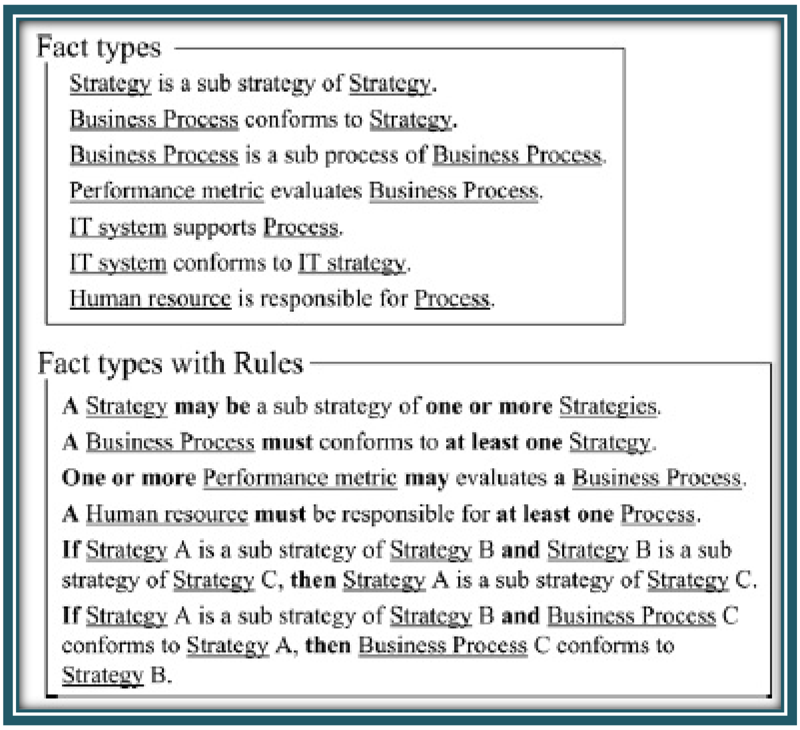


Figure 3-12: Business Model for Fact-based Ontologies (Kang, Lee & Kim, 2010: 3276)

The fact-based approach to ontology rests on concepts with facts, building on fact types, according to rules, which is best illustrated with an example in Figure 3-13 below. When populating these fact types with facts, according to the rules, an enterprise can semantically reason and define the necessary relationships. This does, however, not provide an overview of the relationships at a glance.



Fact types

Strategy is a sub strategy of Strategy.
Business Process conforms to Strategy.
Business Process is a sub process of Business Process.
Performance metric evaluates Business Process.
IT system supports Process.
IT system conforms to IT strategy.
Human resource is responsible for Process.

Fact types with Rules

A Strategy **may be** a sub strategy of **one or more** Strategies.
A Business Process **must** conforms to **at least one** Strategy.
One or more Performance metric **may** evaluates **a** Business Process.
A Human resource **must** be responsible for **at least one** Process.
If Strategy A is a sub strategy of Strategy B **and** Strategy B is a sub strategy of Strategy C, **then** Strategy A is a sub strategy of Strategy C.
If Strategy A is a sub strategy of Strategy B **and** Business Process C conforms to Strategy A, **then** Business Process C conforms to Strategy B.

Figure 3-13: Example of Fact Types (Kang, Lee & Kim, 2010: 3278)

3.4.4.2 DDPO (DOLCE + DnS Plan Ontology)

DDPO is an example of a task ontology/taxonomy. “Task taxonomies can be initially defined from a mathematical viewpoint as graphs that create an ordering over sets of action types” (Gangemi *et al.*, 2005: 5). “The purpose of the DDPO is to specify DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) plans at the abstract level by using First-Order Logic (FOL)” and it contains plans, sub-plans, goals, plan execution, preconditions and post-conditions (Damjanovic, 2010: 497).

DDPO utilises the concepts and relations from DOLCE combined with the ontology of DnS (Descriptions and Situations) to target tasks and develop plans on an abstract level, independent of current resources (Gangemi *et al.*, 2005: 27).

- DOLCE: "aims at capturing the main cognitive categories such as endurants (particulars in space) and perdurants (particulars in time), underlying existing ontologies and human commonsense [sic]" (Damjanovic, 2010: 497). DOLCE uses another two categories, besides endurants and perdurants, namely: qualities and abstracts. Figure 3-14 depicts an example of DOLCE, where the yellow blocks represent the categories and the arrowed lines represent relations. The grey blocks represent the different particulars, dependent on the category. "Particulars can be as varied as possible: in space (e.g. a saxophone) or in time (e.g. a song); physical (e.g. a stone), social (e.g. a company), or mental (e.g. a desire); agentive (e.g. an animal) or non-agentive (e.g. a law); qualities (e.g. the color depending on the pigmentation of a specific eye) or quality spaces (e.g. *sea green* in the Mac palette); substances (e.g. an amount of sand) or systems (e.g. the complex of a car engine, wheels, gears, road, air, driver), etc." (Gangemi *et al.*, 2005: 28).

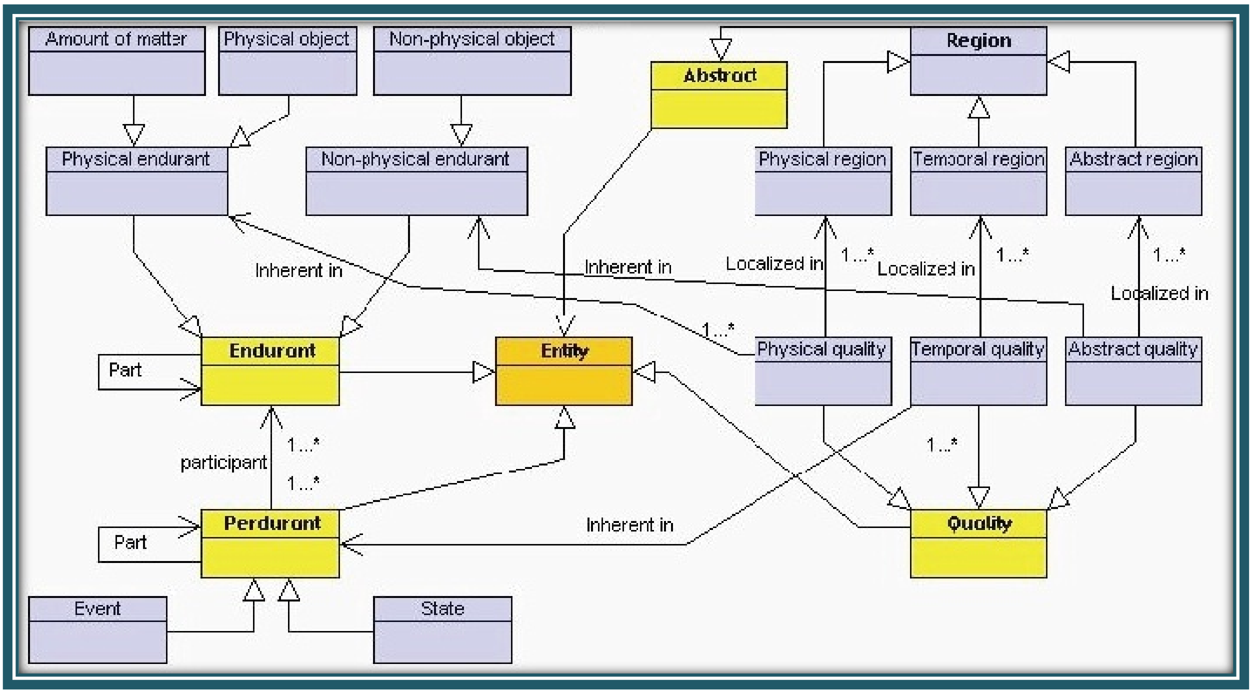


Figure 3-14: Example of DOLCE (Gangemi *et al.*, 2005: 28)

- DnS extension: DOLCE has several extensions, namely: Time Ontology, Space Ontology, Descriptions and Situations Ontology, Plan Ontology, Semiotics and Information Ontology. DDPO makes use of the Descriptions and Situation Ontology. "DnS provides a vocabulary and an axiomatization to type the new individuals, to interrelate them to the existing predicates from another [ontology]" (Gangemi *et al.*, 2005: 29)

3.5 SUMMARY AND CONCLUSION

3.5.1 Summary

The sub-research question answered in this chapter is: *d) What definitions, benefits and models are there for business architecture?*

The question was addressed by first evaluating existing definitions of business architecture and then compiling a working definition thereof. The synthesised working **definition,** as depicted in Figure 3-3, is duplicated here in Figure 3-15 for ease of reference and is written out as:

> *Business architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships and change/innovation of the business strategies/objectives, processes/value chains, capabilities, functions/structure and resources (human and finance) of an enterprise.*

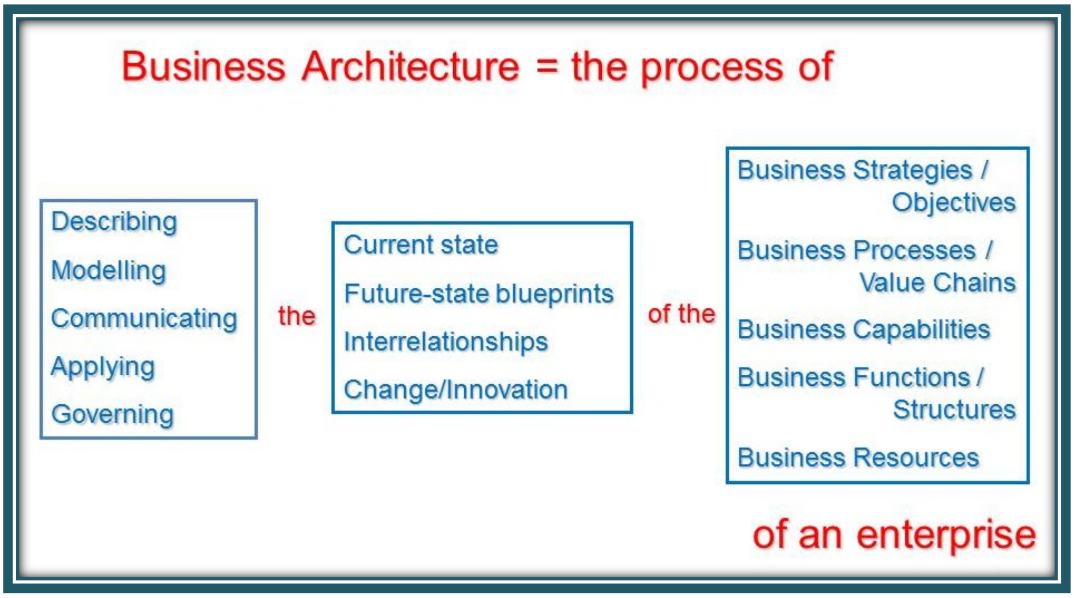


Figure 3-15: Synthesised Definition of Business Architecture

Second, facets of the **role** are, in short, flexibility in change, organisation, creation of business artefacts and gaining insight into and understanding of the business. The **benefits** lie in business architecture's assistance in structuring, guiding, shaping, managing and improving the enterprise.

Third, the different levels of abstraction with regard to **frameworks** and **models** were discussed. Table 3-2 provides a quick overview of the abstraction levels.

TYPES	*EXAMPLES*
EA Frameworks	Business Concepts (Zachman Framework) Business Architecture (TOGAF) DoDAF Business Reference Model (FEAF) Gartner Business Architecture Framework
BA Frameworks	Microsoft Motion Business Architecture Methodology A New Business Architecture for UC Agile Business Process Modelling Framework Business Motivation Model
Modelling Languages	Process Model Relational Model Causal Loop Diagram Object-Oriented Model Pi-Calculus Process Algebra Business Process Model and Notation Capability Model Web Services Description Language (WSDL) Business Process Execution Language (BPEL) Semantics of Business Vocabulary and Business Rules (SBVR) Archimate
Ontologies	Fact-Based Ontologies DDPO (DOLCE + DnS Plan Ontology)

Table 3-2: Overview of Levels of Frameworks and Models (Business Architecture)

3.5.2 Conclusion

Business architecture is an important part of enterprise architecture as it describes the context and environment of the enterprise, which is to be captured by EA. A synthesised definition of business architecture has been provided. Business architecture has specific roles within an enterprise and can provide significant benefits to the enterprise. There are different frameworks, models, ontologies and descriptive languages, which can be utilised to describe an enterprise's business architecture and create architecture artefacts.

Business architecture will play a significant role as the departure point and driver in the creation of an overarching architectural model, interlinking the different architectural domains. The next chapter (Chapter 4) will describe the information architecture layer of enterprise architecture.

4 INFORMATION ARCHITECTURE

4.1 INTRODUCTION

Business architecture was discussed in the previous chapter. The next high-level overview is provided on the architectural domain of information architecture (IA). The subject of the next sub-research question is thus:

e) What definitions, benefits and models are there for information architecture?

Organisations are currently looking for new ways to exploit and leverage their information assets – information architecture is thus becoming an important area of focus in this regard (Newman, Gall & Lapkin, 2008: 1). Information architecture should maintain a balance between the business goals and the user requirements or needs (Rosenfeld & Morville, 2007: 5).

The current status of information architecture is described here by means of a critical analysis of the existing literature, through exploring the definitions, role and benefits of information architecture. The output of this analysis is a proposed integrated definition. Some of the existing frameworks and models are also described briefly. This will provide essential background to create an understanding of the environment and will be used as input in creating a model to facilitate the integration between the different architectural layers or domains.

4.2 DEFINITIONS OF INFORMATION ARCHITECTURE

The terms '*information architecture'* or '*enterprise information architecture'* (EIA) are broad terms that are often misunderstood (Newman, Gall & Lapkin, 2008: 1). Various definitions, existing in the literature, will be explored here in order to be able to compile a working definition of information architecture.

The term '*information'* can bring about the whole debate about the differences between data, information and even knowledge. This issue will not be debated here. For the purpose of this study we are concerned with information of all shapes and sizes including data (transactional and summarised) and metadata. A few examples are books, databases/datasets, journals, music, maps, art works, websites, electronic documents/images/clips/blogs (Rosenfeld & Morville, 2007: 5). Gartner (Handler & Newman, 2005: 4) provides a list of information sources/forms: structured sources, such as databases, transaction systems and applications, as well as unstructured sources like e-mails, documents, rich media and other formats.

The term '*architecture'* has already been defined in paragraph 2.2 in Chapter 2.

4.2.1 Definitions in the Literature

The scope of information architecture ranges from Batley (2007: 4), listing the core concepts as indexing, classification, cataloguing and user-centric designs; to Godinez *et al.* (2010) positioning IA as translating the business requirements into informational strategies and data definitions. Wurman published a milestone definition of an information architect in 1996 (Wurman & Bradford, 1996) and has been widely cited (Haller, 2010: 13; Sack & Hearst, 2001; Toms, 2002: 855):

"1) the individual who organizes the patterns inherent in data, making the complex clear

2) a person who creates the structure or map of information which allows others to find their personal paths to knowledge.

3) the emerging 21st century professional occupation addressing the needs of the age focused upon clarity, human understanding and the science of the organization of information."

This definition is important, albeit of the architect himself, as this was published prior to the definitions of the subject 'information architecture'. Subsequently these definitions of information architecture were selected from the literature and are discussed below in chronological order to highlight the progression of IA. TechTarget (Moon, 2000) published the following definition: "information architecture is the set of ideas about how all information in a given context should be treated philosophically and, in a general way, how it should be organized". They apply this to technical writing as well as more narrowly to web content. It includes the "areas of content management, content distribution or syndicating, and electronic publishing". This description of IA lacks reference to usability, metadata, presentation and navigation and does not refer to links to any of the other architectures.

Sack & Hearst (2001) list the following elements of information architecture:

- “Organization systems
- Labelling systems
- Navigation systems
- Search and indexing systems
- Metaphor systems
- Audience analyses”.

These elements are part of the IA process, but the definition is lacking in content, context, relations to the business and technology architecture and the process nature of architecture.

Brinck, Gergle & Wood (2001: 120) define IA as referring “to the structure or organization of your web site, especially how the different pages of the site relate to one another. It involves such issues as content analysis and planning, organization of the pages, providing cues to help users orient themselves, labeling, search techniques, and navigation design”. The main problem with this definition is that it only refers to websites and not to information in the enterprise as a whole.

Higgins & Hebblethwaite (2006: 4) derived a definition from a number of sources – “Information architecture is the means of providing a structured description of an enterprise’s information, the relationship of this information to business requirements and processes, applications and technology, and the processes and rules which govern it”. This definition addresses the relevant issues on a high level, without mentioning the activities like labelling, navigating and searching information.

Rosenfeld & Morville (2007: 4) defined information architecture, as initially published in 2002 (Rosenfeld & Morville, 2002), as:

1. “The combination of organization, labelling, and navigation schemes within an information system.

2. The structural design of an information space to facilitate task completion and intuitive access to content.
3. The art and science of structuring and classifying websites and intranets to help people find and manage information.
4. An emerging discipline and community of practice focused on bringing principles of design and architecture to the digital landscape."

This definition has been widely used and quoted (Crystal, 2007; Singh, 2011: 152; The Information Architecture Institute, 2007). They also state that information architecture is not graphic design or software development or even usability engineering. This definition is comprehensive, but does not address the relationships to the other architectural domains.

Crystal (2007: 16) describes enterprise information architecture as having four components, *i.e.* organisation, navigation, labelling and searching. Information architecture is thus to "organize information appropriately by creating a hierarchical structure that is comprehensible to users, provide navigation structures that enable users to move through the information space, label categories and groups of content in sensible ways, and design search systems (including retrieval algorithms and metadata) that allow users to search for information effectively". The enterprise wide perspective as well as relationships is missing.

The Information Architecture Institute (2007) defines information architecture as:

"The structural design of shared information environments.

The art and science of organizing and labeling web sites, intranets, online communities and software to support usability and findability.

An emerging community of practice focused on bringing principles of design and architecture to the digital landscape".

Although this definition does imply the process nature of architecture it needs reference to the enterprise context and relations. The activity of searching is also missing.

Gartner (Newman, Gall & Lapkin, 2008: 2) defines IA "as that part of the enterprise architecture process that describes – through a set of requirements, principles and models – the current state, future state, and guidance necessary to flexibly share and exchange information assets to achieve effective enterprise change." Gartner presents a high-level definition, placing IA in context, but lacking detailed reference to the IA activities.

The Business Dictionary (Business Dictionary, 2010) defines IA as a: “set of rules that determine what, and how and where, information will be collected, stored, processed, transmitted, presented and used. On the internet, information architecture means how a website’s content is organized and presented to its users to facilitate navigation and search functions.” This definition covers the principles of architecture as well as the activities for achieving it.

Godinez *et al.* (2010: 29) explain information architecture as: “the framework that defines the information-centric principles, architecture models, standards, and processes that form the basis for making information technology decisions across the enterprise” and “Enterprise Information Architecture translates the business requirements into informational strategies and defines what data components are needed by whom and when in the information supply chain.” These are good high-level definitions, placing IA in the enterprise context, but do not shed light on the activities to achieve IA.

The content of the above definitions was analysed based on the terminology used. The matrix in Table 4-1 provides a summary of terms, synonyms and verbs used. The terms were grouped together under output formats, activities, objects and scope. The group objects were subdivided into goals and information assets. The terms most frequently used are ’structure’ and ‘organises’, followed by ‘principles’, ‘navigation’ and ‘management of content’.

		Authors	Moon	Sack & Hearst	Brinck, Gergle & Wood	Higgins & Hebblethwaite	Rosenfeld & Morville	Crystal	IA Institute	Gartner	Business Dictionary	Godinez *et al.*
Output formats		Systems/structures	o	•	•	•	•	•	•			
		Models								•		•
		Standards/rules				•				•	•	•
		Processes								•		•
		Principles	•			•	•		•	•	o	•
		Frameworks										•
Activities		Organises	•	•	•		•	•	•		•	
		Navigates		•	•		•	•	•		•	
		Labels		•	•		•	•	•			
		Searches		•	•			•			•	
		Classifies					•					
		Presents									•	
		Distributes	•								•	•
		e-Publishes	•		•							
Objects	Goals	Usability			•		•	•	•		o	
		Accessibility (sharing)							•	•		
		Change/innovation								•		
		Audience knowledge		•	o		•					
		Content management	•		•		•	•	•		o	•
	Information Assets	Information space			•		•	•	•			
		Metaphors		•								
		Enterprise information				•						
		Metadata						•				
		Websites	•		•		•		•		•	
		Intranets					•		•		•	
Scope		Current state								•		
		Future state								•		
		Enterprisewide				•						•
		Relation to business				•				o		•
		Relation to IT				•				o		•
		Process approach				•				•		
	o	Implied in definition	•	Mentioned in definition								

Table 4-1: Analysis of Information Architecture Definitions

4.2.2 Synthesised Definition

An synthesised working definition of information architecture was compiled and is illustrated in Figure 4-1: Synthesised Definition of Information Architecture. The basis for this definition is the synthesis of the literature definitions above, as well as the fit within the bigger context of enterprise architecture.

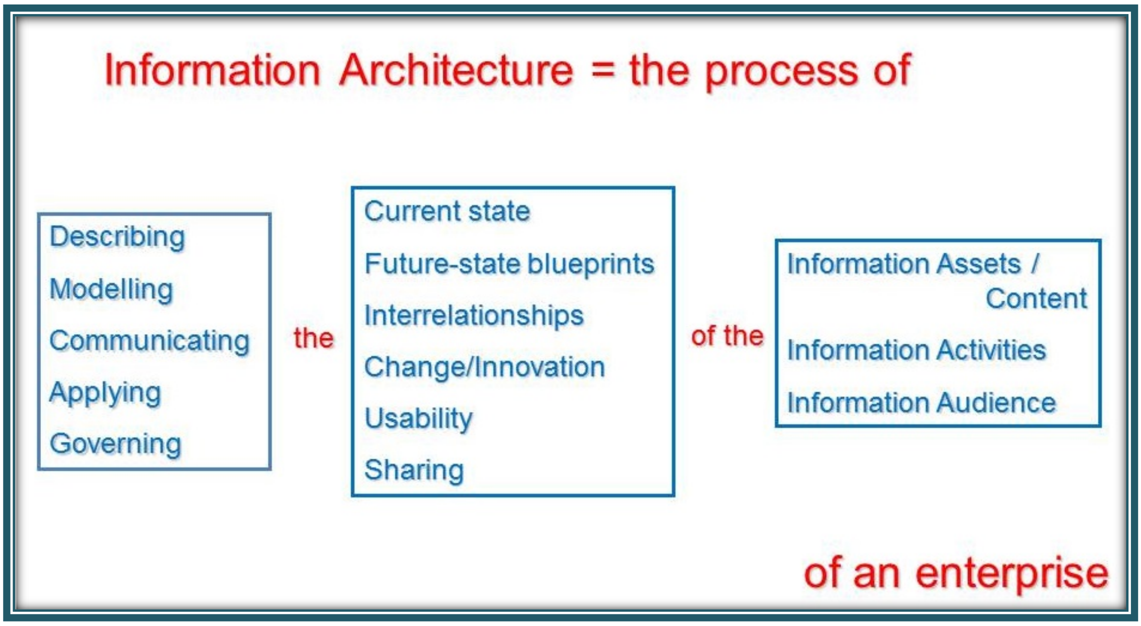


Figure 4-1: Synthesised Definition of Information Architecture

The proposed definition in paragraph format is:

> *Information architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships, change/innovation, usability and sharing of the information assets/content, information activities, and the information audience of an enterprise.*

For a specific project, focus or enterprise a subset of the terms can be used, for example: IA is the process of modelling the future-state blueprints of the information assets of an enterprise.

4.3 THE ROLE AND BENEFITS OF INFORMATION ARCHITECTURE

The value of information architecture to an enterprise can be addressed by investigating the specific role it plays within enterprise architecture as well as the possible benefits. Some of the facets of this **role** are:

- Facilitating task completion,
- intuitive access to content and
- managing information (Rosenfeld & Morville, 2007: 4).

- Achieving a high quality solution, while reducing costs, and
- enabling an enterprise to create, maintain and use information assets throughout their life-cycle (Godinez *et al.*, 2010).

- Providing documents to form the basis of information planning as well as input to information design (Moon, 2000).

- Providing principles and guidelines for consistent implementation of solutions (Godinez *et al.*, 2010).

- Reducing the cost of finding information, finding the wrong information and not finding information at all;
- reducing the cost of maintenance and training,
- improving knowledge sharing, and
- solidifying the enterprise's business strategy (Rosenfeld & Morville, 2007: 344).

- Achieving "a high level of integration of managerial and technical elements" and
- the large-scale integration of enterprise systems (including models, data, information, knowledge and web descriptions) (Neaga & Harding, 2005: 1089).

- Providing a common language set for all units/departments/agencies across the enterprise,
- creating of enterprise wide classification schemes, and
- establishing inter-departmental agreements around information management, data architecture and information sharing (Higgins & Hebblethwaite, 2006: 8).

- Providing competitive advantage because of IA's influence on enterprise agility (Schlier, 2006: 1).

- Enhancing the user experience by providing the basic platform/framework (Haverty, 2002: 840).

- Enabling the enterprise to organise, prioritise and plan information assets in support of strategic business objectives, and
- providing a blueprint to ensure information achieves the objective of the enterprise (Handler & Newman, 2005: 5).

Some of the possible cited **benefits** of information architecture are:

- Increased sales,
- enhanced website experience,
- improved brand loyalty, and
- reduced duplication of effort (Rosenfeld & Morville, 2007: 345).

- Fostered innovation (The Information Architecture Institute, 2007).

- Consistent, repeatable and thus predictable project success, and
- flexible responsiveness to the business process and the enterprise (Godinez *et al.*, 2010).

- Consistently described artefacts,
- facilitated communication between business, information and IT architects, and
- a basis for an information asset inventory (Higgins & Hebblethwaite, 2006: 9).

Information architecture thus has an important **role** within an enterprise to manage its information assets by using common languages/structures/classifications to enhance integration, solutions, cost reduction, access, task completion and competitive advantage. Information architecture can produce **benefits** mainly in improving access to information (including sales and brand loyalty), reducing duplication of effort and enhancing communication between business and IT.

4.4 EXISTING FRAMEWORKS AND MODELS

There are existing frameworks and models to describe the artefacts of information architecture. Artefacts can include wireframes, blueprints, controlled vocabularies and metadata schemas (Rosenfeld & Morville, 2007: 15) as well as user profiles, use case models, system context diagrams, service qualities and logical/component/operational model diagrams (Godinez *et al.*, 2010). Sack & Hearst (2001) also view documents such as newspapers, books, city access guides and websites as artefacts of information architecture. Newman & Landay (2000) describe several additional artefacts: sitemaps, storyboards, schematics, specifications, mock-ups, prototypes, guidelines, presentations and documents.

In order to explore the existing frameworks and models of information architecture, different levels of abstraction need to be addressed:

- First, the component/view of the enterprise architecture frameworks, relating to information architecture will be discussed, followed by other information architecture-specific frameworks.
- Second, a number of models, which can be used for providing content to the framework, are touched upon.
- Third, ontologies will be discussed briefly to express the relationships between elements.
- Fourth, some descriptive languages, which can be used to express the content of the frameworks or models, are listed.

4.4.1 Information Components of Enterprise Architecture Frameworks

Some of the enterprise architecture frameworks discussed in paragraph 2.7 have specific information architecture components and will be discussed briefly here.

4.4.1.1 Zachman Framework

The Zachman Framework does not have a specific row or column labelled ‘information’, but it has a row dedicated to the system concepts from an architect’s perspective. This view is composed of:

- the logical data model – the 'what question';
- the logical process model / data flow diagram – the 'how question';
- the network schematic model / distributed systems architecture – the 'where question';
- the workflow and interface design – the 'who question';
- the logical dynamics model / state transition dependency diagram – the 'when question'; and
- the business rules model – the 'why question' (Hokel, 2010: 4; Macgregor, 2008).

The Zachman Framework also has a column labelled 'data' (Domagalski, 2008). As per definition this column intersects with the following rows:

- the scope or contextual layer;
- the business model or conceptual layer;
- the system model or logical layer, as described above;
- the technology model or physical layer; and
- the detailed representations layer.

4.4.1.2 TOGAF

Information systems architecture (application architecture) is phase C of TOGAF's Architecture Development Method (ADM), following after business architecture. The objectives are to define the major kinds of relevant application systems necessary to process the data and support the specific enterprise and what those applications need to do in order to manage data and to present information to the human and computer actors in the enterprise. "The applications are not described as computer systems, but as logical groups of capabilities that manage the data objects in the Data Architecture and support the business functions in the Business Architecture" (The Open Group, 2009c: 127).

A variety of modelling tools can be employed to do the modelling (The Open Group, 2009c: 127):

- Generic business models relevant to the industry/vertical of the enterprise, for example, healthcare, transportation or finance.

- Application models relevant to common high-level business functions, such as electronic commerce or supply chain management.
- The Open Group also has a Reference Model for Integrated Information Infrastructure (III-RM) that focuses on the application-level components and services necessary to provide an integrated information infrastructure.

4.4.1.3 DoDAF

DoDAF does not contain a section specifically called ‘information architecture’, but the ‘Systems and Services View’ (SV) has relevance. “The SV captures system, service, and interconnection functionality providing for, or supporting, operational activities” (USA. Department of Defence, 2007a: 1-8) and has an extensive list of useful artefacts, for example:

- Systems functionality descriptions
- Service functionality descriptions
- Operational Activity to Systems Function Traceability Matrix
- Operational Activity to Systems Traceability Matrix
- Operational Activity to Service Traceability Matrix
- Systems Interface Description
- Services Interface Description
- Systems Communications Description
- Services Communications Description
- Systems-Systems Matrix
- Services-Systems Matrix
- Services-Services Matrix
- Systems Functionality Description
- Services Functionality Description
- Systems Data Exchange Matrix
- Systems Performance Parameters Matrix
- Services Performance Parameters Matrix
- Systems Rules Model
- Services Rules Model (USA. Department of Defence, 2007a: 1-10) .

A variety of modelling tools can be employed to do the modelling (USA. Department of Defence, 2007b: 2-4):

- Structured Models – to provide a process driven hierarchical decomposition of functional processes.
- Object-Orientated Architectures – to describe the operational need in the context of its use, based on the concepts of data abstraction.
- Activity-Based Methodology (ABM) – to enable development and analysis of integrated architectures using a disciplined approach.
- Architecture Specification Model (ASM) – to provide a common set of semantics.

4.4.1.4 FEAF

FEAF makes use of the Data Reference Model (DRM) for creating the information architecture part of enterprise architecture. DRM is a model "describing, at an aggregate level, the data and information that support program and business line operations". DRM's goal is to provide business-focused data standardisation and cross-agency information exchanges (USA. FEA Program Management Office, 2004).

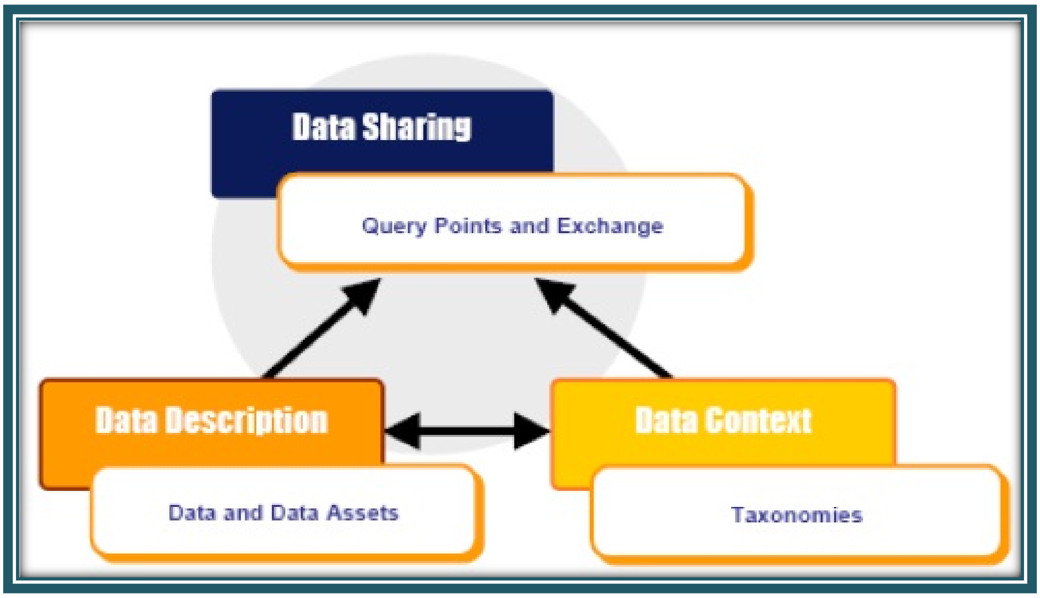


Figure 4-2: The structure of DRM (Executive Office of the President of the United States, 2007: 8)

The DRM provides a standard means by which data may be described, categorised, and shared, as depicted in Figure 4-2, consisting of three standardisation areas:

- Data Description – to describe data uniformly, thereby supporting its discovery and sharing.
- Data Context – to facilitate discovery of data through an approach to the categorisation of data according to taxonomies.
- Data Sharing – to support the access and exchange of data, enabled by capabilities provided by both the Data Context and Data Description standardisation areas (Executive Office of the President of the United States, 2007: 7).

4.4.1.5 Gartner Information Architecture Framework

Gartner (Blechar, 2007: 7) claims that, with the adoption of Service Oriented Architectures (SOA as described in 4.4.2.1), an information architecture and infrastructure management will become increasingly important. The inclusion of integration methods and tools needs to be addressed other than with a tactical and fragmented approach. This evolving of IA and infrastructure is depicted in Figure 4-3.

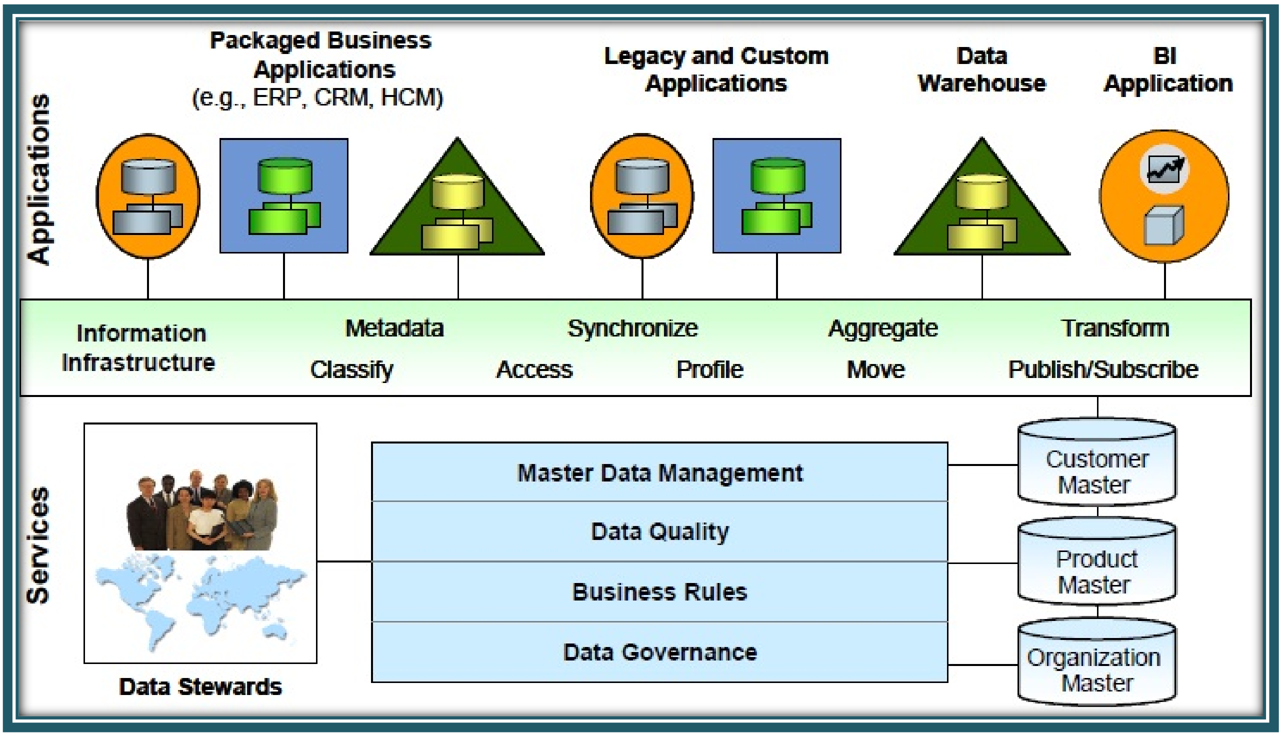


Figure 4-3: Gartner: Evolving the Information Architecture and Infrastructure (Blechar, 2007: 7)

The different applications and application types can be integrated by different actions to enable the enterprise to deliver and govern services. The IA will provide "the policies, models and rules for designing and implementing effective software and data services and databases" (Blechar, 2007: 7).

4.4.2 Information Architecture Frameworks

There are also frameworks, other than the above EA components, available for information architecture. A few of these frameworks will be discussed briefly below, starting with Service-Oriented Architectures.

4.4.2.1 Service-Oriented Architectures

The design of SOA was driven by the need for solutions, that are flexible, computing platform independent and that can be integrated. "The basic idea of SOA paradigm is that a system is designed and implemented using loosely coupled software services with defined interfaces that can be accessed without knowledge of their implementation platform" (Rehan & Akyuz, 2010: 2607). It thus provides a means for integration across diverse systems, by using concepts like web services, process orchestration and enterprise services as well as open standards like Simple Object Access Protocol (SOAP), WSDL (Web Services Description Language) and XML (Extensible Markup Language).

According to Brown (2008: 178), SOA is "aimed at representing business processes through choreographed sequences of services realized through reusable components". This is now the focus of many enterprises for the future of agility in their software. Gartner (Blechar, 2007) adds to this by emphasising that SOA exposes data issues to more people, places and processes, which cannot be sorted out without an information management and architecture focus. This also brings about new challenges as depicted in Figure 4-4.

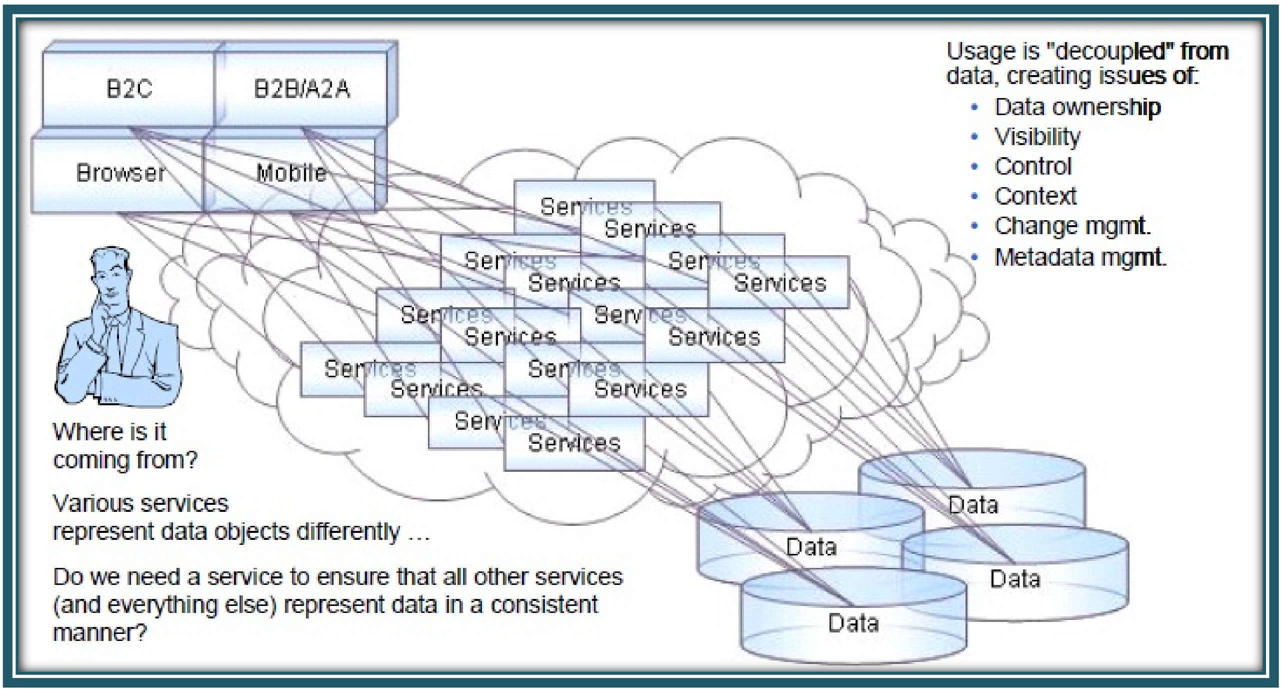


Figure 4-4: Gartner: SOA challenges (Blechar, 2007: 5)

Both the applications and the infrastructure must support the SOA principles in order to implement a SOA successfully within an enterprise. For the applications this implies the creation of service interfaces directly or through the use of adaptors. For the infrastructure this implies routing and delivering capabilities as well as the support for safe substitution of one implementation by another without an effect on the client (Papazoglou & Van den Heuvel, 2007: 396).

4.4.2.2 <u>'Facets are Fundamental' Framework (FaF)</u>

Crystal (2007) proposes the 'Facets are Fundamental' framework in an attempt to address three issues with other IA frameworks:

- topic-centricity – website characteristics *versus* user's needs;
- facets are supplemental – instead of fundamental; and
- conflating organisation and representation – unclear distinctions between organisation and navigation.

This framework argues that faceted classification should form a more fundamental part of the IA process. Faceted classification is the starting point of the IA development process. It also distinguishes between facets and attributes – this leads to employing non-topical methods for organising and representing information. This will enhance other frameworks and make IA more flexible (Crystal, 2007).

Facets are defined in the Cambridge Dictionary (2011) as: “one part of a subject, situation, etc. that has many parts” and in the Oxford Dictionary (2011) as: “one side of something many-sided” and “a particular aspect or feature of something”. Yee, *et al.* (2003) describe faceted classification as “composed of orthogonal sets of categories”, that can be either flat or hierarchical and either contains a single value or multiple values. Each facet can have schemes and structures beneath them. Crystal (2007) adds to this that an (information) resource can have facets as well as attributes, where attributes are tight syntactic data, while facets are loose pieces of assigned information. Facets are thus prone to interpretation and influences by culture, politics and environment.

FaF begins with developing facets and attributes (rather than a hierarchy or set of categories), using these to classify the information space and then developing representations of the information space to provide access to structured information. By using facets the information design is more balanced and has “broader conceptions of relevance and pertinence” (Crystal, 2007). Examples of facets are audience, location, task and genre. The outputs, for websites, are typically wireframes for:

- topics – containing a space for tasks, a space for different audiences and topical information or links;
- audiences – containing a space for relevant tasks, a space for documents of specific format or genre and navigation; and
- content – containing a space for the content, a space for possible relevant links and navigation.

The FaF framework is used to enhance the existing frameworks by utilising facets as the fundamental starting point of organising and labelling information resources.

4.4.2.3 The Common Knowledge Enterprise Model

The next IA framework, proposed by Neaga & Harding (2005: 1098), is based on knowledge discovery (KD) and data-mining. They argue that "extending the existing enterprise modeling and integration architectures and environments to incorporate KD and data-mining (DM) systems could significantly contribute to improving the decision-making process and business performance" (Neaga & Harding, 2005: 1090). Information systems, which follow reference architectures and use standards, can be incorporated in the IA design. The same is applicable to knowledge discovery and data-mining applications adhering to standards. This framework, 'The Common Knowledge Enterprise Model', makes use of existing models from the Object Management Group (OMG) like the Model Driven Architecture (MDA) model and the Common Warehouse MetaModel (CWM).

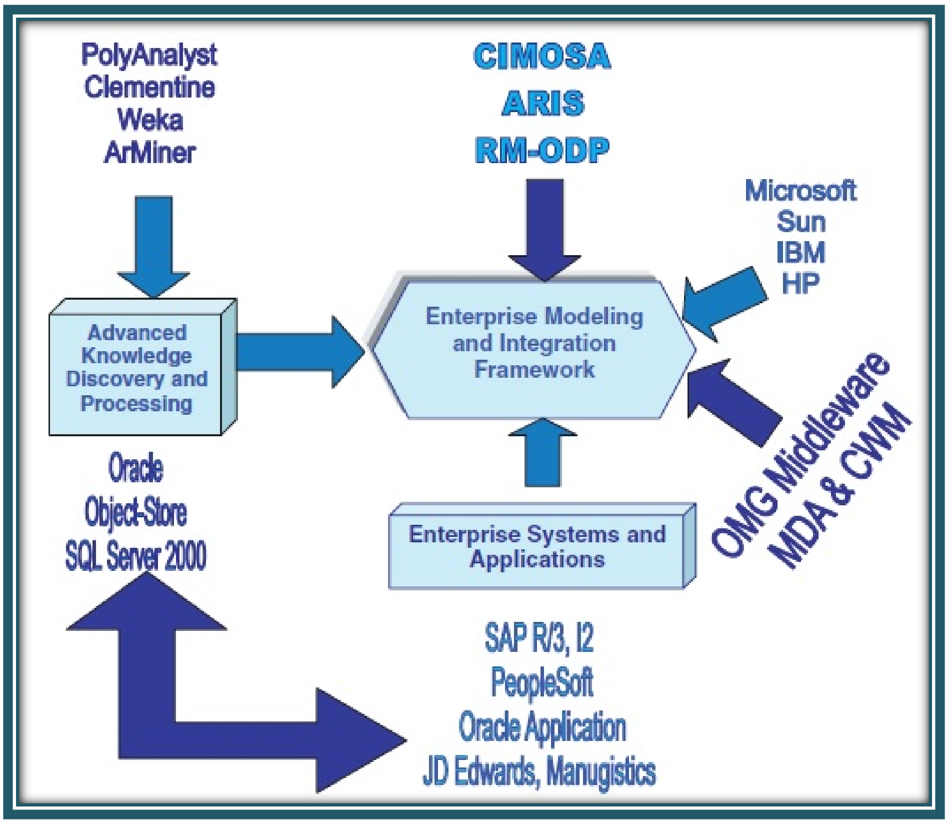


Figure 4-5: The Common Knowledge Enterprise Model (Neaga & Harding, 2005: 1099)

Figure 4-5 depicts the common knowledge enterprise model and its design, development and implementation. The "framework defines a unified environment for integration" (Neaga & Harding, 2005: 1098) of:

- knowledge discovery and data-mining systems – for example, PolyAnalist and Clementine;
- source libraries of programs – for example, Weka and ArMiner;
- enterprise systems – for example, SAP (Systems, Applications & Products in Data Processing) and PeopleSoft;
- existing reference architecture – for example, MDA and CWM; and
- existing architecture models – for example, CIMOSA and RM-ODP.

The common knowledge enterprise model thus supports platform-independent developing of standard information systems for collaborating by making use of Unified Modelling Language (UML) or XML and integrating with a variety of existing products, systems and frameworks.

4.4.2.4 Information Architecture Abstract Model

The Queensland Government (Higgins & Hebblethwaite, 2006) attempted to create a framework to harmonise the different terminologies and semantics, to enable cross-agency information management principles, to provide governmentwide classification schemes, to enable consistent distribution of artefacts and to facilitate compliance. This is part of their Government Enterprise Architecture. Their view of the scope of information architecture is depicted in Figure 4-6 – this includes information management, information content and the information portfolio.

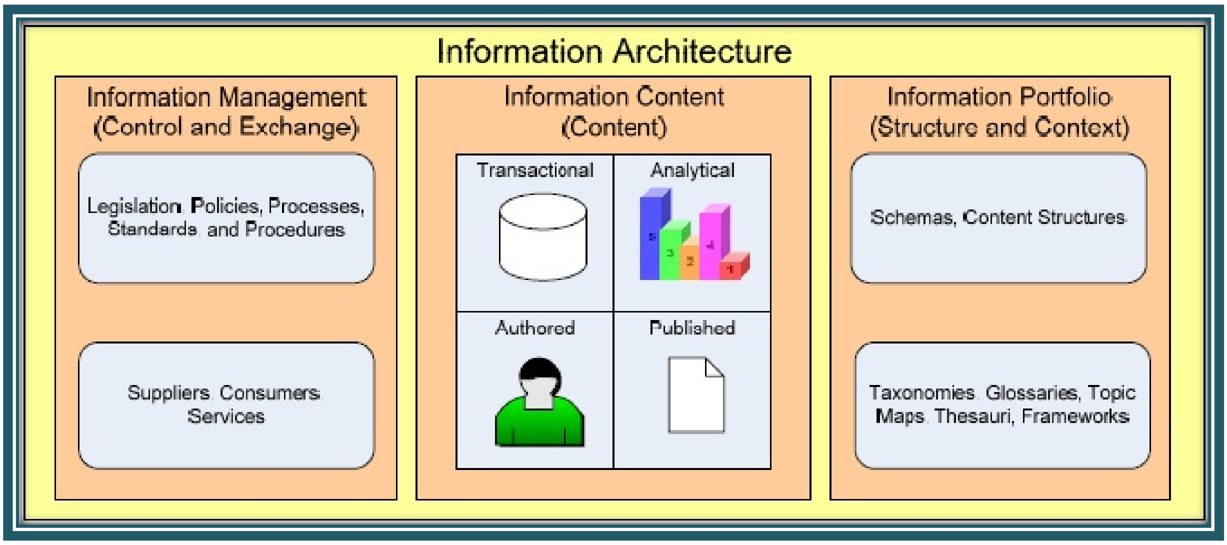


Figure 4-6: Scope of Information Architecture (Higgins & Hebblethwaite, 2006: 4)

The information architecture abstract model is based on the Data Reference Model (DRM) as part of FEAF. It describes the three major concepts, as illustrated above, and indicates the relationships between them. “In doing so it provides a whole-of-government pattern for information architecture and presents the minimal level of detail necessary to convey the meaning behind each of the major concepts” (Higgins & Hebblethwaite, 2006: 9). It contains a diagrammatic model (see Figure 4-7) and a detailed list of definitions of the concepts, containing examples, relationships, alternative terms and the relevant sub-area.

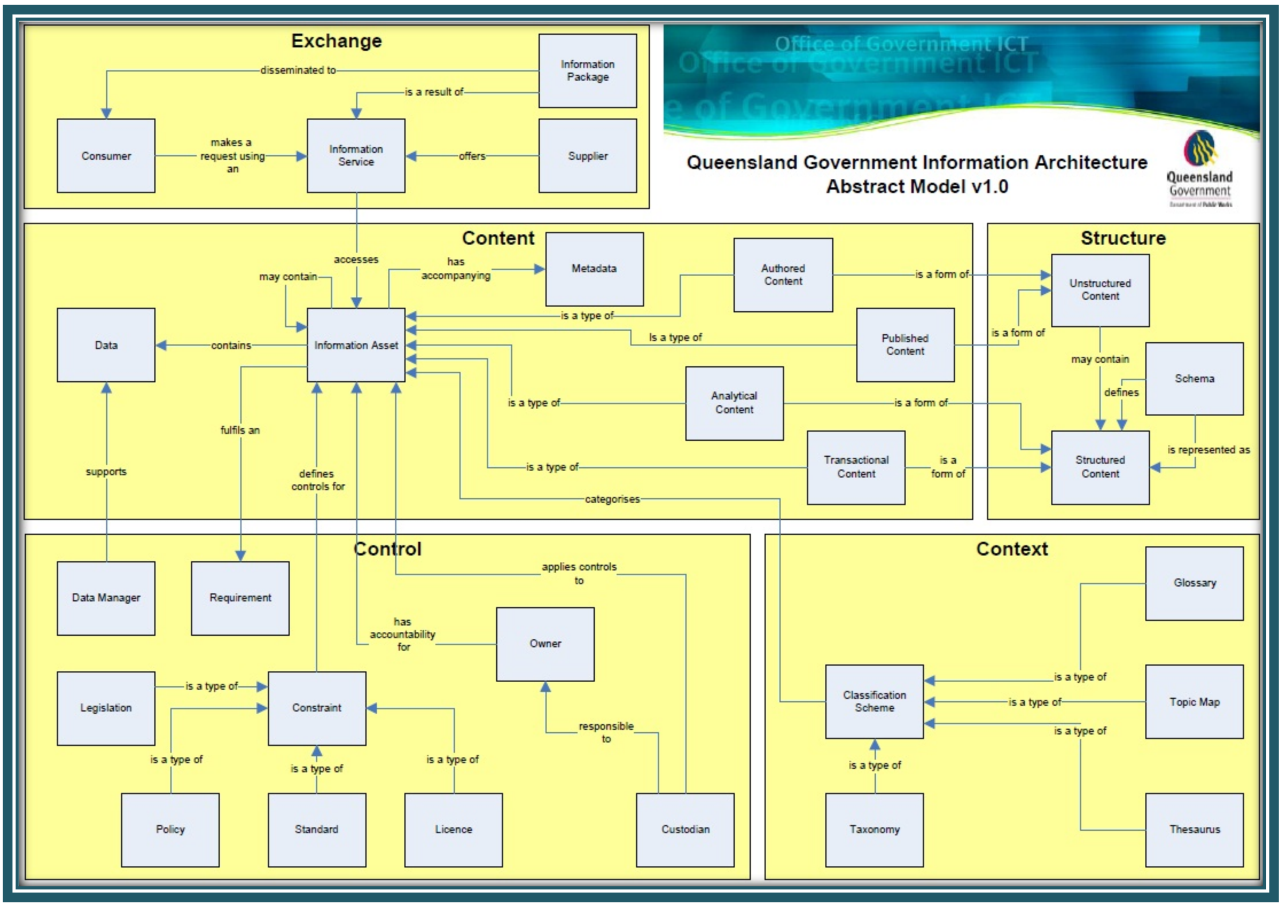


Figure 4-7: Queensland Government Information Architecture Abstract Model (Higgins & Hebblethwaite, 2006: 11)

4.4.2.5 Strategic Information Architecture (SIA)

Pai & Lee (2005: 153) propose the strategic information architecture framework as a possible solution to gain flexibility to handle rapidly changing environments and enterprise requirements. The proposed SIA is depicted in Figure 4-8. SIA focuses on integrating the information systems (IS), information technology and information management (IM) strategies of an enterprise.

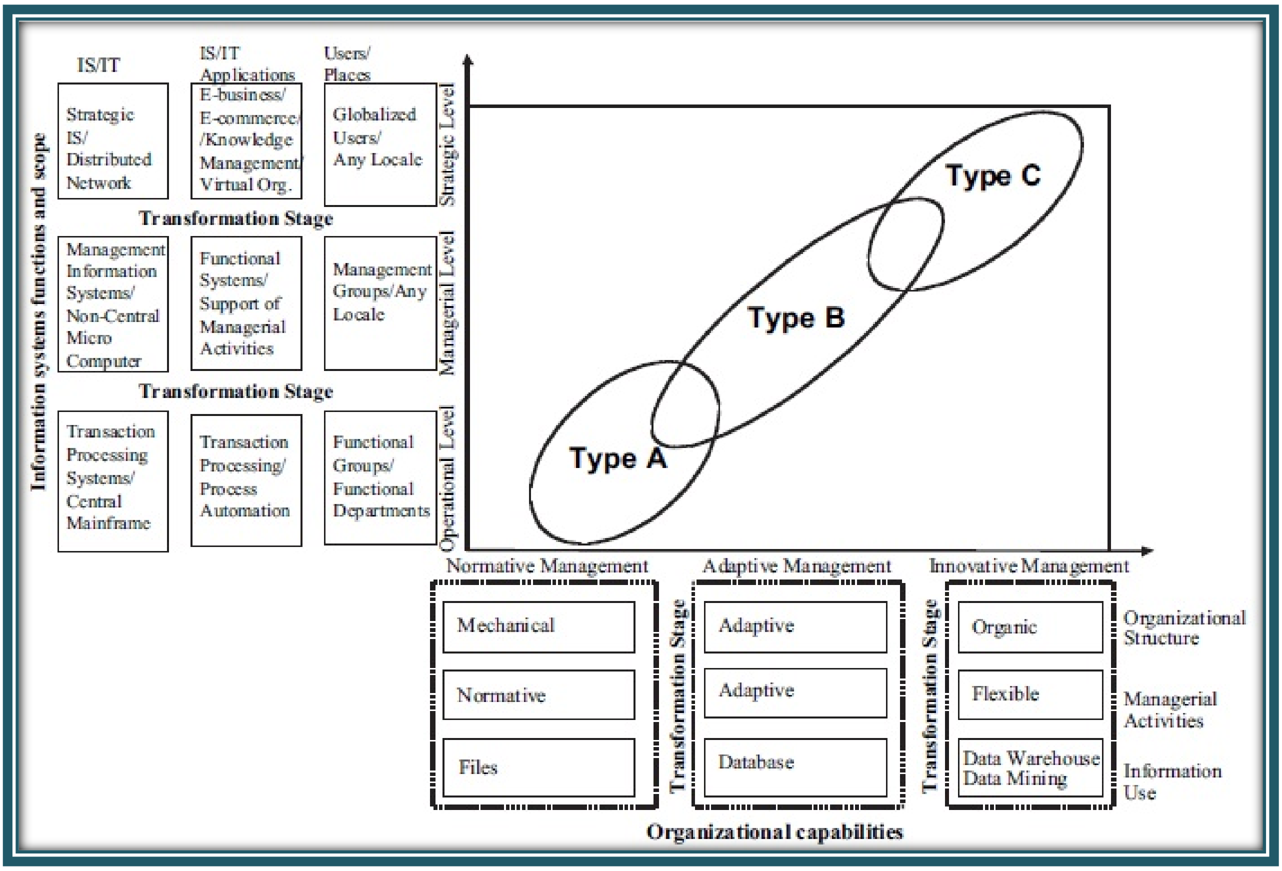


Figure 4-8: The SIA integrating IS, IT and IM strategy (Pai & Lee, 2005: 154)

The framework consists of two axes:

- The horizontal axis – indicating the business conditions and capabilities, covering:
  - o the enterprise structure – including mechanical, adaptive and organic structures,
  - o managerial activities – including normative, adaptive and flexible managerial activities,
  - o information use – through the different eras of IS development, as well as
  - o normative, adaptive and innovative management styles.
- The vertical axis – indicating information systems functions and scope, covering:
  - o the operational level – IS/applications to upgrade the operation efficiency of the enterprise,
  - o the managerial level – IS/applications to be applied to supplement the management of the enterprise,

- o the strategic level – IS/applications to improve the compatibility of the enterprise, as well as
- o the users and locale orientation.

The framework also indicates the fitted (type A, B and C) situation of SIA:

- Type A – showing the operational IS functions and scope,
- Type B – representing the enterprise capability which must move toward adoptive management when IS have reached management level,
- Type C – emphasising the strategic IS applications which must cope with innovative management.

"The capabilities of an enterprise must be compatible when IS functions and scope change" otherwise it will result in non-fitted situations (Pai & Lee, 2005: 156).

4.4.2.6 <u>The Evernden Eight</u>

Evernden & Evernden (2003: 28) describe eight essential factors "that need to be taken into account when forming an information architecture". These factors arose from years of experience in architecting, the practice of key methodologies and theory from diverse disciplines. These are depicted in Figure 4-9.

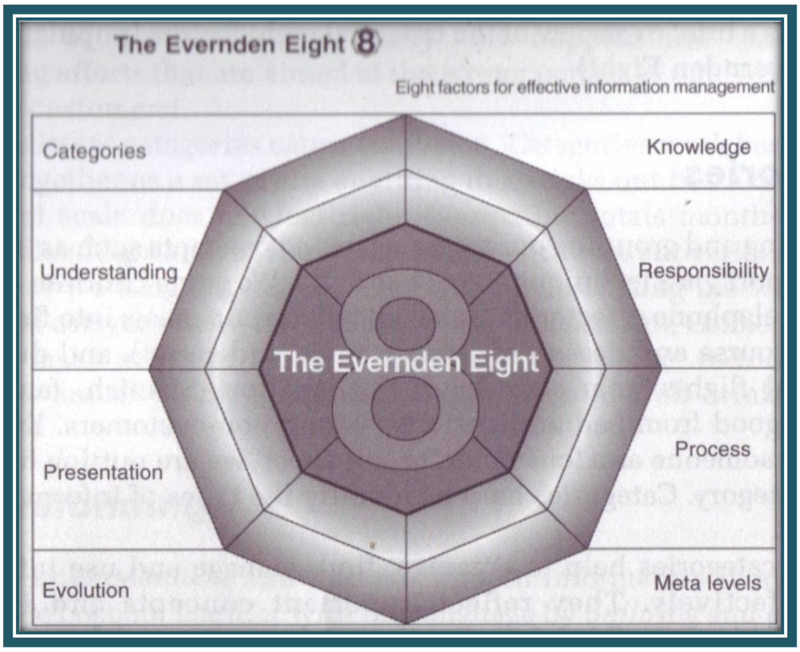


Figure 4-9: Evernden Eight (Evernden & Evernden, 2003: 29)

The Evernden eight are (Evernden & Evernden, 2003: 30):

- Categories – to organise, find, manage and use information more effectively.
- Understanding – to find meaning in information by descriptions, examples, guidelines, procedures and definitions.
- Presentation – to enhance usage and understanding of information by presenting in various formats, styles and level of detail.
- Evolution – to keep information relevant and up to date by qualifying it with time, frequency and reviews.
- Knowledge – to reveal and share knowledge by deciding which knowledge to codify and which not.
- Responsibility – to manage the information assets by establishing roles, accountabilities and responsibilities.
- Process – to analyse the process of using information in order to improve decision-making, to simplify flows, to develop value chains and to maximise the use of information.
- Metalevels – to create language, grammar, templates and patterns to provide information about information.

4.4.3 Modelling Languages

According to Pai & Lee (2005: 151) there are different approaches to the development of information architecture, *i.e.* technology-oriented approaches, strategic-oriented approaches, scenario-based approach and the world-class enterprises approach.

Models, as reasoned in paragraph 2.8, are used to fill in the different structures and sections of a framework. The generic models (process model, relational model, causal loop diagram and object-oriented model), as discussed in paragraph 2.8, can also be used for information architecture.

There are other models that can also be used to fill in the different sections or views of a framework:

- Metadata-Modelling – addressing data integration (the technical aggregation of data from various systems) and cognitive integration (the synthesis of this data to create new understanding). Multiple perspectives are needed, *i.e.* a content perspective, a user perspective, an enterprise perspective and a technical perspective (Hert *et al.*, 2007: 1268).

- Information Interaction Model – "crosses the 'no-man's land' between user and computer" (Toms, 2002: 855) by including the user, content and system and their interactivity. Users bring their human information processing capabilities, the systems bring dynamic artificial intelligence processes, and the content a knowledge presentation containing content items and structure (Toms, 2002: 859).
- Generic Model of EIA for a Public Institution – including a data collection model, an information core, a metadata management module, a management model (operational, tactical, strategic), and a data-mining and knowledge discovery module (Bologa, Faur & Ghisoiu, 2010: 23).
- Instructional Design Models – useful for user profiling, usability testing, carrying out task audits, performing walk-throughs, and storyboarding (Ford & Mott, 2007: 336).
- Conceptual Model – based on the second row of the Zachman Framework, defines each information subject on a high level with categories/types, internal entities, attributes, channels and external entities (Domagalski, 2009).
- Usability Models – useful for conducting user testing and to integrate user feedback (Ford & Mott, 2007: 336),
- Extended Influence Diagrams – start with determining the utility node and then consist of an iterative process of identifying nodes that affect or define other nodes. This continues until every node's relation to the utility node is clear. These relations are depicted in an extended influence diagram (Johnson *et al.*, 2006).

A number of descriptive languages are listed here. Most of them are based on or integrate with the Extensible Markup Language (XML). XML is a simple, very flexible text format aimed at large-scale electronic publishing, which is now used for the exchange of a wide variety of data on the Web and elsewhere (Quin, 2012). XML uses plain Unicode text and predictable-looking tags (Mertz, 2002). In XML effective information object creation is best done by using a document model to declare an object's elements and hierarchical structure, like:

- Document Type Definition (DTD) – this language is particularly compact and describes the structure of XML documents effectively.

- XML Schema Definition (XSD) – has been developed by the W3C (World Wide Web Consortium) and offers an effective and flexible approach to constructing schemas.
- RELAX NG (Regular Language for XML New Generation) – simplifies the complexities associated with XSDs and is considered to be more robust, more succinct, and simpler to use than XML (Ford & Mott, 2007: 334).
- Darwin Information Typing Architecture (DITA) – organises each chunk of information by applying a type to it. DITA first uses a generic topic type and then three core information types, *i.e.* concept, task and reference. The additional types can then be created with flexibility although guided by a set of rules. The structural relationships are also indicated (Verbert *et al.*, 2004: 717).
- PYX – is an open source simplified XML format using line-oriented format for representing and processing of XML documents with common text tools. PYX is not actually XML but represents information in XML documents in an easier format (Mertz, 2002). PYX describes the information communicated by an XML parser to an XML application (McGrath, 2000).
- Schema for Object-Oriented XML (SOX) – is a schema language for defining the syntactic structure and partial semantics of XML document types. SOX is more extended than DTD, but it still decreases the complexity of interoperation among heterogeneous applications by mapping XML data structures with abstractions and relationships (Davidson *et al.*, 1999).
- Sorta Like Python (SLiP) – "a quick, alternative shorthand syntax for creating and editing XML data by hand" (Sweeney, 2002).
- Simple XML (SXML) – is a modified version of XML with the purpose of minimising the size effects of XML, like size of overheads and slow implementation speed. This is especially aimed at the mobile learning environments (Shi & Zhang, 2009).

4.4.4 <u>Ontologies</u>

Ontology, as described in more detail in paragraph 3.4.4, uses the origins of a concept to classify concepts into a schema and to portray the interrelations. A few of the ontologies in literature are summarised below.

- Fractal approach – When there are heterogeneous domain ontologies as well as complex enterprise and information systems architectures, the fractal approach to information systems is applicable in order to support a flexible change process from the business domain to the technology domain. "Fractal approach allows applying various systems development methods at different administrative levels of the business organization and thus can accelerate the IS development process and change management during the systems maintenance" (Kirikova, 2009: 135).
- Ontology-Driven Development – The Nottingham University Hospitals developed a clinical information model. "The model is represented as an ontology, coded in OWL/XML and is itself built upon an ontology-based information architecture" (Ontology Web Language) (Chelsom *et al.*, 2011: 1). This model has been implemented using an open source toolkit and can be used to generate the runtime configuration and operational data structures for a clinical information system. Ontologies are used for the diagnosis classifications at different levels, a data dictionary of clinical findings, clinical coding and the configuration and runtime data structures (Chelsom *et al.*, 2011: 2).
- Ontologies for SOA – The Open Group developed ontologies for SOA for use by business people as well as architects. "It defines the concepts, terminology and semantics of SOA in both business and technical terms" to enhance communication and understanding and contributes towards a model-driven implementation (Macgregor, 2008: 31).

4.5 SUMMARY AND CONCLUSION

4.5.1 Summary

The sub-research question answered in this chapter is: *e) What definitions, benefits and models are there for information architecture?*

The question was addressed by first evaluating existing definitions of information architecture and then compiling a working definition thereof. The synthesised working **definition,** as depicted in Figure 4-1, is duplicated here in Figure 4-10 for ease of reference and is written out as:

Information architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships, change/innovation, usability and sharing of the information assets/content, information activities, and the information audience of an enterprise.

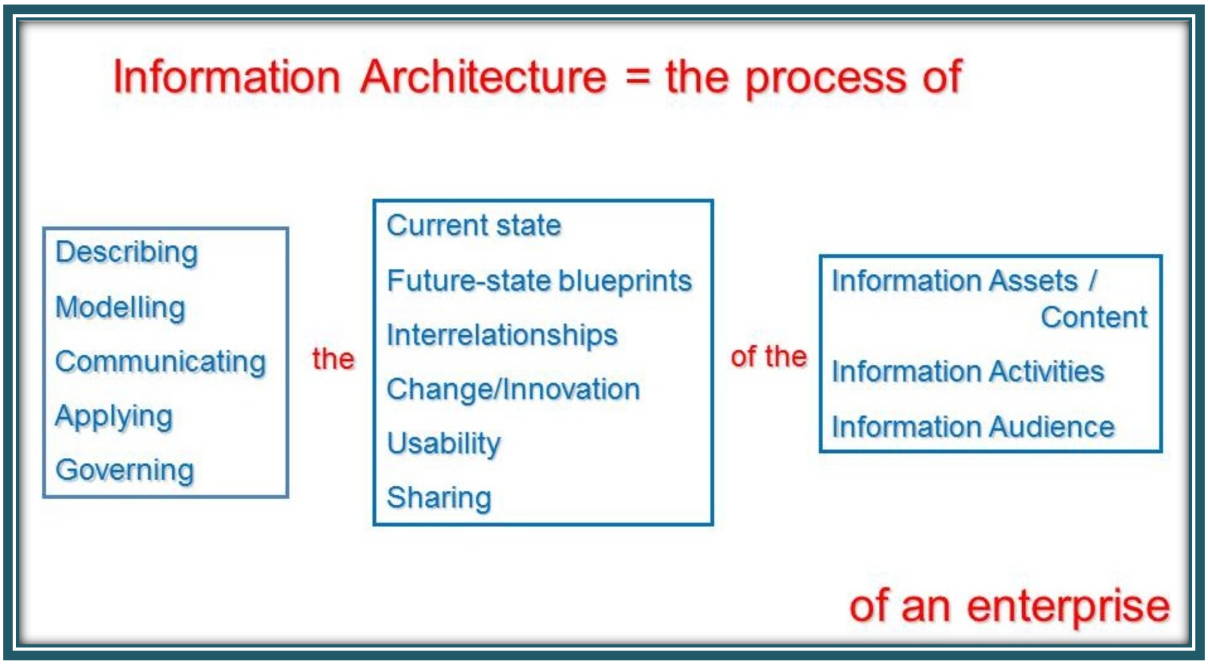


Figure 4-10: Synthesised Definition of Information Architecture

Second, facets of the **role,** within an enterprise, are to manage its information assets by using common languages/structures/classifications to enhance integration, solutions, cost reduction, access, task completion and competitive advantage. The **benefits** lie in information architecture's assistance in improving access to information (including sales and brand loyalty), reducing duplication of effort and enhanced communication between business and IT.

Third, the different levels of abstraction and examples with regards to **frameworks** and **models** were discussed. Table 4-2 provides a quick overview of the abstraction levels.

TYPES	*EXAMPLES*
EA Frameworks	Systems Concepts (Zachman Framework) Information Systems Architecture (TOGAF) Systems & Services View (DoDAF) Data Reference Model (FEAF) Gartner Information Architecture Framework
IA Frameworks	Services Orientated Architectures 'Facets are Fundamental' Framework The Common Knowledge Enterprise Model Information Architecture Abstract Model Strategic Information Architecture The Evernden Eight
Modelling Languages	Metadata-Modelling Information Interactive Model Generic Model for EIA for a Public Institution Instructional Design Model Usability Model Conceptual Model Extended Influence Diagrams

Table 4-2: Overview of Levels of Frameworks and Models (Information Architecture)

There are also a variety of descriptive languages available, most of which are related to XML, for example, XSD, DITA, SOX and SXML.

4.5.2 Conclusion

Information architecture is an important part of successful enterprise architecture as it describes and governs all the different information sources and systems that are essential for achieving business goals and engagement with the users/clients. Flexibility and agility are important gains from implementing information architecture successfully.

Information architecture is included in most of the well-known enterprise architecture frameworks, but there are also specific information architecture frameworks, models, ontologies and descriptive languages for implementing IA.

Information architecture will play a significant role in the creation of an overarching architectural model, as it forms a link between the business drivers and the implementation thereof with information technology. The next chapter (Chapter 5) will describe the technical architectural domain of enterprise architecture.

5 TECHNOLOGY ARCHITECTURE

5.1 INTRODUCTION

'Information architecture' was discussed in the previous chapter. The next high-level overview is provided on the architectural domain of 'technology architecture' (TA). The subject of the next sub-research question is thus:

f) What definitions, benefits and models are there for technology architecture?

IT is playing a more central role in the transformation of enterprises in facing the challenges of the 21th century. "IT should be an essential means to achieve business value, as well as to create new organizational forms with an increased ability to innovate, compete and cooperate" (Pessi, Magoulas & Hugoson, 2011: 53). This leads to three significant trends, *i.e.* the use of IT is continuously influencing every area of social and business life, IT investment issues are now of interest to the entire enterprise, and IT management has become increasingly complex and difficult. This emphasises the importance of technology architecture within enterprise architecture.

Chanopas, Krairit & Khang (2006: 633) state that IT is constantly and quickly evolving with improved IT products and services being released regularly and that IT infrastructure is a long-term asset, has shareholder value and represents the long-term options of an enterprise.

The current status of technology architecture is hereby described by means of a critical analysis of existing literature, through exploring the definitions, role and benefits of technology architecture. Some of the existing frameworks and models are also described briefly. This will provide essential background to create an understanding of the environment and will be used as input in creating a model to facilitate the integration between the different architectural layers or domains.

5.2 DEFINITIONS OF TECHNOLOGY ARCHITECTURE

The term ‘architecture’ has been used in a variety of ways connected to IT. It ranges from ethical concepts to the physical makeup of vendors’ products, the structure of information, the delivery of technology and also the technical management of the IT solutions (Perks & Beveridge, 2004: 1).

Robertson (2010: 1) indicates that there are many who still use the term ‘IT architecture’ synonymously with ‘enterprise architecture’, which is incorrect as ‘IT architecture’ describes a more limited technical architecture focus.

Definitions of technology architecture in the literature will be explored in order to be able to compile a working definition of technology architecture. The terms ‘*technology architecture’* (TA), ‘*information technology architecture’* (ITA) and ‘*enterprise technology architecture’* (ETA) are used interchangeably by some authors in this domain (Robertson, 2010: 2).

5.2.1 <u>Definitions in the Literature</u>

The following definitions of technology architecture were chosen from the literature and are discussed chronologically below.

In 2001 the Open Group started to define IT architecture as a “formal description of an information technology system, organized in a way that supports reasoning about the structural properties of the system” (The Open Group, 2009c). This definition is cited, *inter alia* by (Bredemeyer & Malan, 2001: 8; Open Security Architecture, s.a.). The problem with this definition is that it shows no EA or enterprise context and does not address the necessary activities.

Huhta (2002) does not supply a formulated definition, but describes technology architecture as a ‘tool’ to link with the business strategy and appropriate business information, by communicating, governing, renewing and innovating the technology. This includes a technology strategy; published policies, practices and methodologies;

defined standards and a documented technology overview. This definition touches on most of the aspects of technology architecture, but TA is more than just a 'tool'.

Carbone (2004: 9) provides the following definition: "'Architecture' means the set of plans that describe how all parts of the IT infrastructure need to behave to support the enterprise needs and goals. It includes all the data required to run the enterprise and the functions, technology, and people that create, access, use, or transform that data into information – and ultimately, knowledge for the business." This definition is lacking in addressing the activities as well as the specific IT related objects.

Perks & Beveridge (2004: 5) describe technical architecture as the enterprise capability that defines "the technical and governance platform on which an organization builds its IT systems to support business benefit." This definition is lacking in addressing the activities, specific objects and the relationship to information architecture.

The Open Group (2004: 1) describes IT architecture as the technical foundation of an effective IT strategy and specifically "an IT architecture defines the components or building blocks that make up the overall information system. It provides a plan from which products can be procured, and systems developed that will work together to implement the overall system. It thus enables you to manage your IT investment in a way that meets the needs of your business." This definition is more comprehensive than the previous one but does not indicate any specific objects, like standards or policies.

Tash (2006) sees TA as the orderly arrangement of technology parts (the individual investments that collectively make up an organisation's technology portfolio) in order to bridge the gap between IT and business people by providing a common framework. This definition lacks detail, for example, specific objects and activities involved in TA.

Winter & Fischer (2007: 2) provide the following description: "The technology architecture represents the fundamental organization of computing/telecommunications hardware and networks. A broad range of design and evolution principles from computer science is available for this layer too." This definition is vague and does not address the enterprise/EA context, the artefacts or the purpose/output formats of TA.

The Open Security Architecture Group (s.a.) suggests the following combined definition: "A set of design artefacts, that are relevant for describing an object such that it can be produced to requirements (quality) as well as maintained over the period of its useful life (change). The design artefact describe [sic] the structure of components, their interrelationships, and the principles and guidelines governing their design and evolution over time." This definition omits the EA context of technology architecture.

Jin, Kung & Peng (2010: 293) view technology architecture as: "Describing the capabilities which support business, data and logic software and hardware of application services deployed, including IT infrastructure, middleware, network, communication, processing, standard." This definition does not contain the TA activities.

The following definition of enterprise technology architecture was summarised from a Gartner report (Burton & Allega, 2011: 66): the "enterprise technology architecture (ETA) viewpoint of enterprise architecture (EA) defines the reusable standards and guidelines for the use of technologies and products", the way it interoperates with the other architectural viewpoints, the technical patterns, services and modelling. It provides drive to decisions on the future state within the business context as well as gap analysis of the current state. This definition should also include the output formats and TA activities.

The content of the above definitions was analysed based on the terminology used. The matrix in Table 5-1 provides a summary of terms, synonyms and verbs used. The terms were grouped together under output formats, activities, objects and scope. The group objects were subdivided into organisational and IT components. The terms most frequently used are: IT components and relation to business followed by IT systems and IT infrastructure. Further terms frequently used are foundation/framework, governance and relation to information (architecture).

		Authors	Open Group (2009)	Huhta	Carbone	Perks & Beveridge	Open Group (2004)	Tash	Winter & Fischer	OSA	Jin, King & Peng	Burton & Allega
Output formats		descriptions	•							•	•	
		Tools		•								
		foundation/framework/ platform				•	•	•	o			
		artefacts								•		
		set of plans			•		•					
Activities		organises						•	•			
		communicates		•								
		governs/manages		•		•	•			•		
		innovates/renews		•					o	o		
		procures/produces					•			•		
		develops/designs					•		•	•		
		maintains								•		
Objects	Management	policies/guidelines		•						•		•
		relations								•		
		methodologies/practices		•								
		standards/principles		•						•		•
		capabilities									•	
	IT	systems	•		o	•	•				•	
		infrastructure			•	o			•		•	o
		components/properties	•		•		•	•	o	•		
		Strategy		•			•					
		overview/portfolio		•				•				
		investment					•	•				
Scope		current state										•
		future state										•
		enterprisewide				o	o					o
		relation to business		•	o	•	•	•			•	•
		relation to information		•	•						•	•
		process approach										o
	o	Implied in definition	•	Mentioned in definition								

Table 5-1: Analysis of Technology Architecture Definitions

5.2.2 Synthesised Definition

Based on a synthesis of the literature definitions above, a working definition for technology architecture was compiled and is illustrated in Figure 5-1.

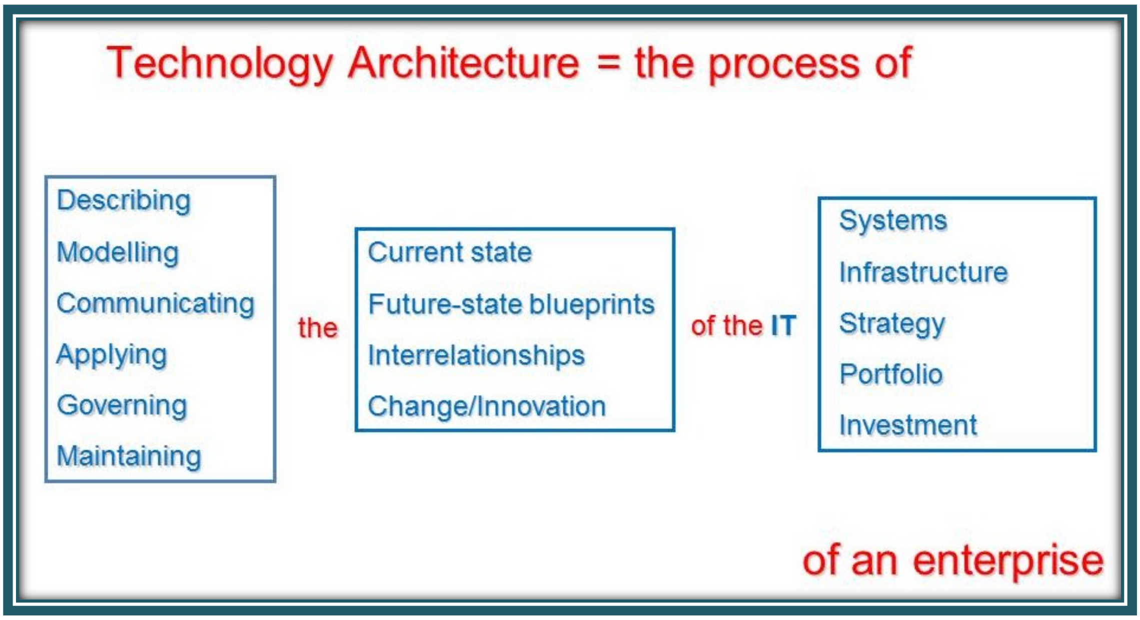


Figure 5-1: Synthesised Definition of Technology Architecture

The proposed definition in paragraph format is:

> *Technology architecture is the process of describing, modelling, communicating, applying, governing and maintaining the current state, future-state blueprints, interrelationships and change/innovation of the IT systems, infrastructure, strategy, portfolio and investment of an enterprise.*

A subset or time-based focused definition can be extracted from the different blocks, for example: technology architecture is the process of modelling the future-state blueprints of the IT infrastructure of an enterprise.

5.3 THE ROLE AND BENEFITS OF TECHNOLOGY ARCHITECTURE

The technology architectural domain has a specific contribution and role to fulfil within the enterprise architecture stack.

Technology architecture has a purpose and plays a specific role within the enterprise. Some of the facets of this **role** are:

- The improving of IT management practice and
- the managing of IT investments (Pessi, Magoulas & Hugoson, 2011: 61).

- The planning of IT (Carbone, 2004: 2; Orr, 2003: 25).

- Finding out what IT assets there are and how they are related to one another, and
- "defining which issues and technologies are currently critical and which ones are likely to become critical (or be phased out)" (Orr, 2003: 26).

- Describing the qualities of an enterprise's infrastructure/systems, that are not evident when the functional components are put next to one another (Open Security Architecture, s.a.).

- Providing agility in decision-making, and
- the ability to respond quickly to an opportunity or threat (Huhta, 2002).

- Describing and maintaining the integrity of the hardware, software and infrastructure environment (Perks & Beveridge, 2004: 16).

- Modelling enterprise artefacts and their relationships (Sousa *et al.*, 2009: 74).

- Achieving competitive advantage, and
- managing innovation and business efficiency (The Open Group, 2004: 1).

- Expediting design of new systems and extensions to existing systems,
- reducing costs of new applications and infrastructure,
- enabling easier interoperability of multiple systems,
- minimising data redundancy, and
- reducing risk (Roberts, 2002: 2).

- Facilitating smart investment decisions, and
- providing the right tools to fit the business requirements (Wiseman, 2007: 7).

Some of the **benefits** of properly applied technology architecture are:

- Reduced cost, and
- increased enterprise application design and development (Radhakrishnan, 2006).

- Sustainable systems,
- dependable systems,
- scalable systems, and
- systems performing according to criteria (Open Security Architecture, s.a.).

- Improved IT decision-making,
- improved business operations,
- improved business-IT alignment, and
- guided business innovation (Huhta, 2002).

- More effective IT operations,
- better return on technology investment,
- improved procurement,
- flexibility for growth and restructuring, and
- faster time-to-market (Perks & Beveridge, 2004: 6; The Open Group, 2004: 2).

Technology architecture thus has an important role within an enterprise to govern IT investments and assets, describing the IT plans, infrastructure and interrelations, and enabling innovation and flexibility. Technology architecture can produce benefits by reducing cost, improving decision-making, operational effectiveness, business alignment and providing growth in flexibility.

5.4 EXISTING FRAMEWORKS AND MODELS

There are existing frameworks and models to describe the artefacts of technology architecture. Artefacts can vary depending on the framework or model used. Examples of artefacts are blueprints and schematic representations of systems, objects and models (Sousa *et al.*, 2009: 84), functional requirements, non-functional requirements, technology architecture principles, policies, standards, guidelines and specifications

(The Open Group, 2009c: 144), principles and inventories (data, applications, platforms, people/processes) (Carbone, 2004: 85).

In order to explore the existing frameworks and models of technology architecture, different levels of abstraction need to be addressed:

- First, the component/view of the enterprise architecture frameworks, relating to technology architecture, will be discussed first, followed by other specific technology architecture frameworks.
- Second, a number of models, which can be used for providing content to the framework, are touched upon.
- Third, ontologies will be discussed briefly to express the relationships between elements.
- Fourth, some descriptive languages, which can be used to express the content of the frameworks or models, are listed.

5.4.1 Technology Components of Enterprise Architecture Frameworks

Some of the enterprise architecture frameworks discussed in 2.7 have specific technology architecture components or views and will be discussed briefly here.

5.4.1.1 Zachman Framework

The Zachman Framework was discussed in paragraph 2.7.1 and has three rows (transformations) relevant to technology architecture. The first relevant row is the fourth row, namely technology, containing specifications or the engineer perspective. The different columns (communication interrogatives) address (Hokel, 2010: 4):

- Technology, What – physical data model (inventory).
- Technology, How – application process specification (process).
- Technology, Where – technology network specification (network).
- Technology, Who – workflow and interface specification (organisation).
- Technology, When – execution timing specification (timing).
- Technology, Why – business rules enforcement specification (motivation).

The second relevant row is the fifth row, namely components, containing configuration or the technician perspective. The different columns (communication interrogatives) address (Hokel, 2010: 4):

- Component, What – database component configuration (inventory).
- Component, How – program component configuration (process).
- Component, Where – network component configuration (network).
- Component, Who – workflow and interface component configuration (organisation).
- Component, When – execution control component configuration (timing).
- Component, Why – business rule enforcement component configuration (motivation).

The third applicable row is the sixth or last row, namely operations, containing instantiations or the workers' perspective. The different columns (communication interrogatives) address (Hokel, 2010: 4):

- Operations, What – operational databases (inventory).
- Operations, How – operational applications (process).
- Operations, Where – operational network (network).
- Operations, Who – operational workflow (organisation).
- Operations, When – operational events (timing).
- Operations, Why – operational rules (motivation).

5.4.1.2 TOGAF – Technology Architecture

TOGAF is described in paragraph 2.7.2. Technology architecture is described in phase D (see Figure 5-2) of TOGAF's Architecture Development Method (ADM), following after Information Systems Architecture (Chase, 2006: 3). The purpose of technology architecture is to map the components of the other architectural domains/views into a set of technology components and into technology platforms. It will define baseline and target views of the technology portfolio and the roadmap towards the target. "As Technology Architecture defines the physical realization of an architectural solution, it has strong links to implementation and migration planning" and it also "completes the set of architectural information and therefore supports cost assessment for particular migration scenarios" (The Open Group, 2009c: 138).

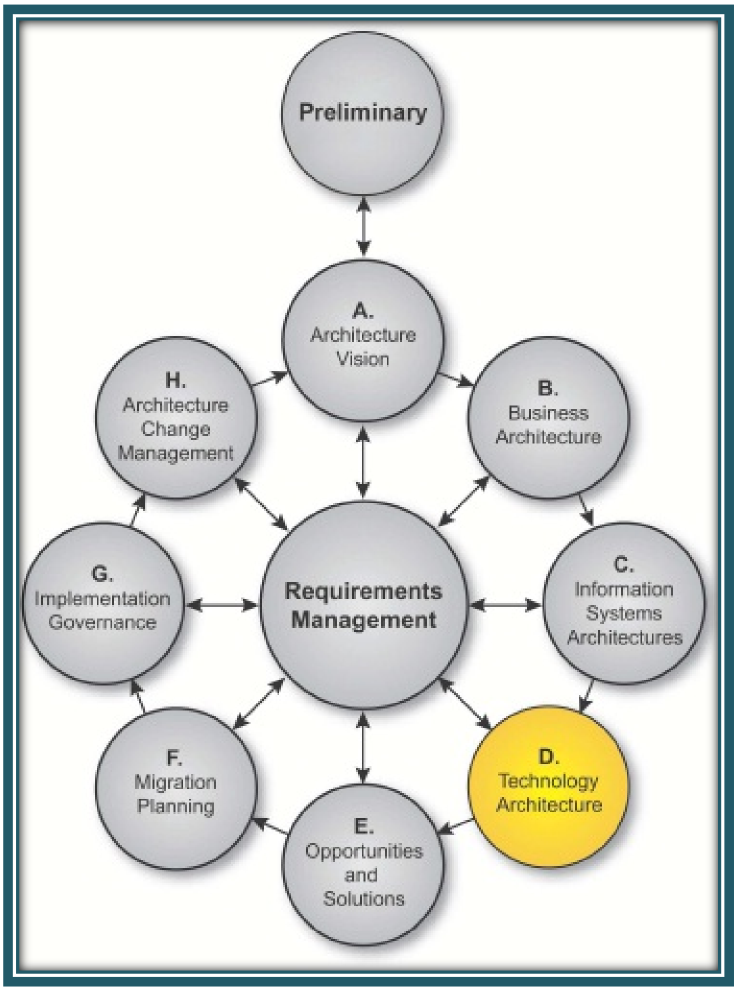


Figure 5-2: TOGAF: Technology Architecture (The Open Group, 2009c: 137)

A variety of technology architecture resources is available under this framework, for example:

- documented IT services in the IT repository or IT service catalogue,
- TOGAF Technical Reference Model (TRM),
- industry (vertical sector) relevant generic technology models, and
- technology models relevant to Common Systems Architectures (The Open Group, 2009c: 138).

5.4.1.3 DoDAF – Technical Standards View

DoDAF, as discussed in paragraph 2.7.3, has a specific view called the 'Technical Standards View' (TV). The TV is the basic set of rules to govern the arrangement, interaction, and interdependence of the systems. The purpose of this view is to ensure that a system satisfies a specified set of operational requirements. It also provides the technical guidelines for systems implementation "upon which engineering specifications are based, common building blocks are established, and product lines are developed. It includes a collection of the technical standards, implementation conventions, standards options, rules, and criteria that can be organized into profile(s) that govern systems and system or service elements for a given architecture" (USA. Department of Defence, 2007a: 1-9).

The 'Technical Standards Profile' and the 'Technical Standards Forecast' are two of the relevant framework products that are available within DoDAF (USA. Department of Defence, 2007a: 1-11).

5.4.1.4 FEAF – Technical Reference Model

FEAF (see paragraph 2.7.4) makes use of the Technical Reference Model (TRM): "Component-driven, technical framework used to identify the standards, specifications, and technologies that support and enable the delivery of service components and capabilities" (USA. FEA Program Management Office, 2004: 7). It is *inter alia* responsible for the service components interfaces, interoperability and technology recommendations.

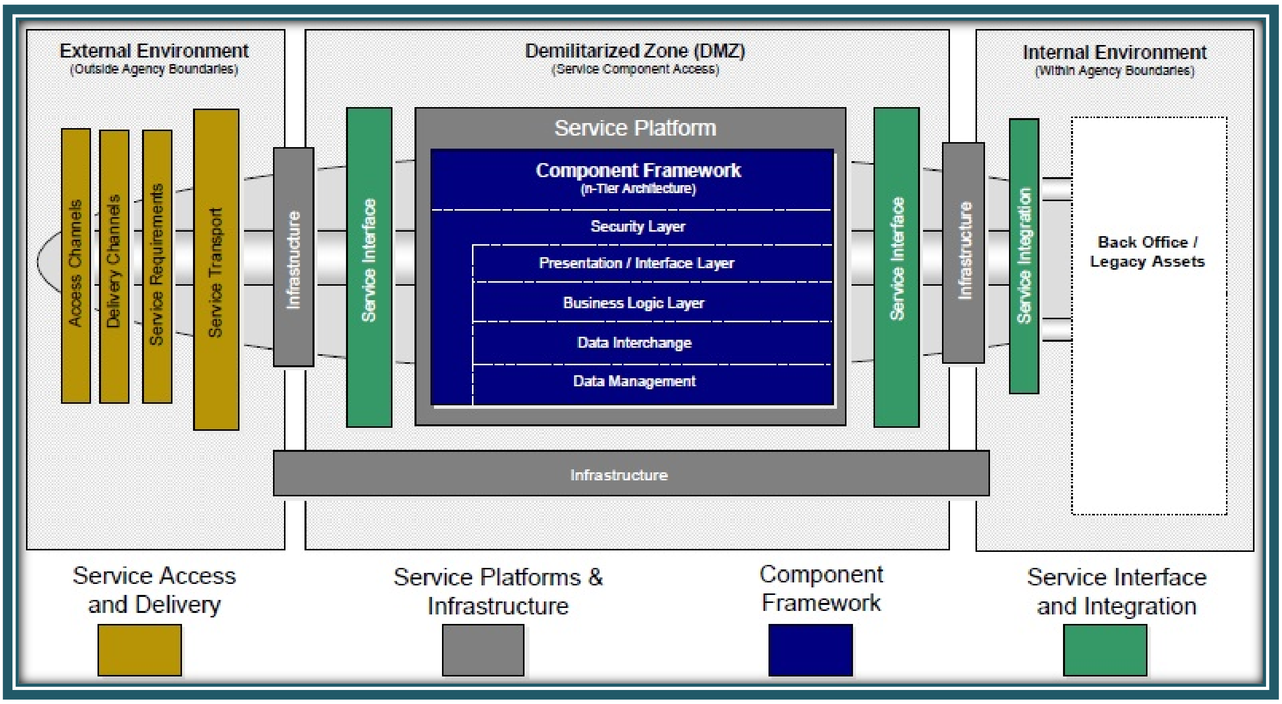


Figure 5-3: FEA's TRM (USA. FEA Program Management Office, 2004: 19)

The TRM, as depicted in Figure 5-3, consists of:

- the external environment – for example: access channels, delivery channels, service requirements and service transport;
- the demilitarised zone (DMZ) – for example: service interface, service platform and component framework;
- the internal environment – back office, legacy assets and service integration;
- the infrastructure – as linkages between the above three (USA. FEA Program Management Office, 2004: 19).

5.4.1.5 <u>Gartner IT Architecture Guideline Framework</u>

The IT Architecture Guideline Framework is intended to gain the support and understanding of the business staff, and to become the basic structure for a more detailed expansion to provide suitable technical guidance for the IT staff. The framework is depicted in Figure 5-4. It has a simple format which should reveal any conflicts. Some cells in the matrix may be more critical or relevant than others, depending on the enterprise's current issues and focus (Rosser, 2002: 1).

IT Architectural Guidelines Content Matrix

Subject Area (Domain)	Level of Specifications				
	Business Need/Goal	Principles	Processes	Industry Standards	Buy List
Data					
Application • Domain			Reference Models		
Service • Desktop • Enterprise • Integration					
Platform • Processor • Network					

Figure 5-4: Gartner's IT Architecture Guideline Framework (Rosser, 2002: 2)

The subject areas (or domains) are in the vertical dimension (Rosser, 2002: 2):

- Data – for example, database products, data-modelling standards and data access tools.
- Application – for example, architecture guidelines for different classes of applications.
- Services:
  - Desktop Services – for example, e-mail and groupware.
  - Enterprise Services – for example, electronic data interchange and security standards.
  - Integration Services – for example, rapid data exchange across diverse applications.
- Platforms:
  - Processor platforms – for example, hardware for desktop, server and hosting environments.
  - Network platforms – for example, storage and network equipment and their operating systems.

The level of specification is in the horizontal dimension (Rosser, 2002: 3):

- Business need/goal – for example, "the 'personality' or 'character' of the architecture".
- Principles – for example, the values and beliefs of the IT enterprise.
- Processes – for example, the user requirement documentation, programming styles and quality assurance methods.
- Industry Standards – for example, special fixed measures applicable to the specific industry.
- Buy List – the first four columns will create the context in which to construct and approve purchases.

This model is not used as such anymore and was expanded as described by Robertson (2005: 4) with:

- technical component catalogue templates – descriptions of the infrastructure components or bricks including life cycle stages (emerging trends, mainstream standards, containment targets and retirement targets);

- technical domain architecture templates – descriptions of the definition of the domain, an overview, the relevant technology trends, design principles, relations to other domains;
- technical pattern model – the use of standardised blueprints for delivery that facilitates the reuse and repetition of designs, processes and delivery plans; and
- technical services – defining shared technical services, for example, network services or web hosting.

These key technology viewpoint organising concepts are depicted in Figure 5-5 and are critical for effectively managing complexity.

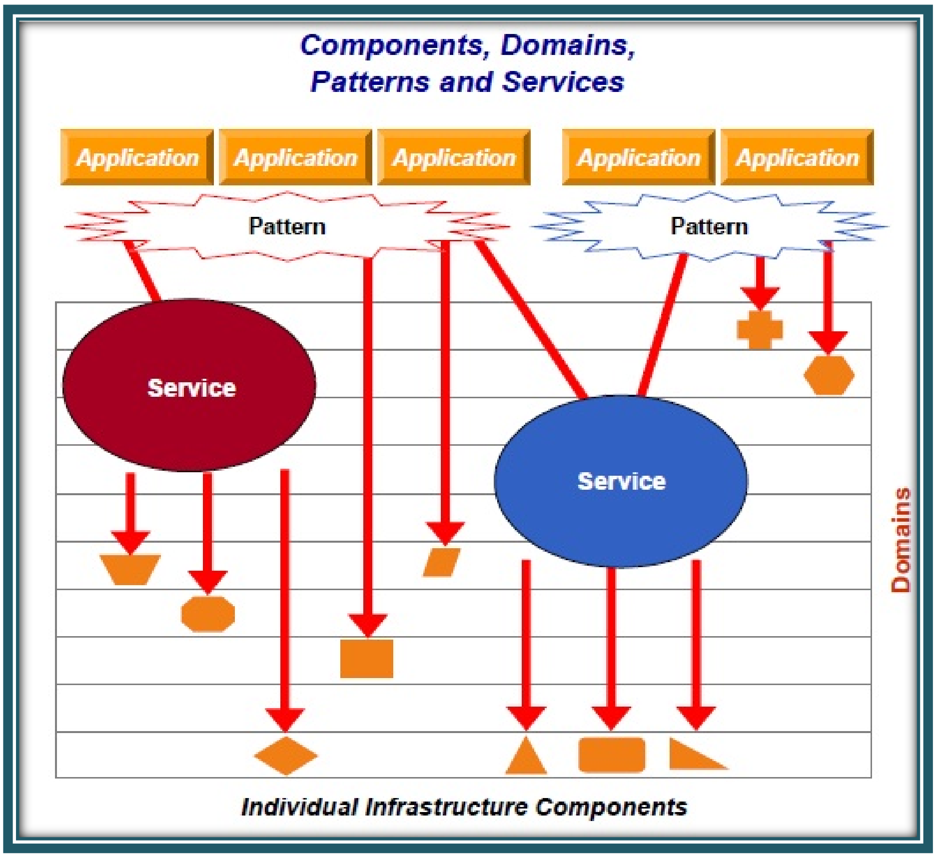


Figure 5-5: Gartner TA Organising Components (Robertson, 2005: 19)

Robertson (2006: 1) from Gartner advocates "a four-slide view of the technical domain's future state, the current state, the gap analysis between current and future state, and the migration plan to move from the current state toward the future state". This approach enables a simple but comprehensive TA description and process by describing the:

- current state – including current trends, principles, industry standards, components and links to other domains;
- future state – including future trends, principles, industry standards, components and links to other domains;
- gap analysis – indicating the changes between the current and the future state; and
- migration plan – including a high-level set of migration projects to consolidate, implement and finalise choices (Robertson, 2006: 7).

5.4.2 Technology Architecture Frameworks

There are other technology architecture frameworks that are not part of the most used enterprise architecture frameworks. Some possible frameworks will be briefly described here, including among others the Strategic Technology Architecture Roadmap, and the WSDOT Architecture Reference Model.

5.4.2.1 IT Framework (Carbone)

Carbone (2004: 10) created a framework, called the IT Framework, to provide a road-map for creating artefacts of IT architecture as well as their relationships that will ultimately enable an enterprise to fulfil its business goals. It also includes best practices to enable the actualisation of the architecture. This framework, as illustrated in Figure 5-6, is intentionally simple and limited in scope to provide a sufficient roadmap to create the necessary artefacts for an IT plan.

	Principles	Models	Inventory	Standards
Data				
Function				
Platform				
People/ Process				

Figure 5-6: IT Framework (Carbone, 2004: 47)

The rows in the framework address the IT components, *i.e.*:

- Data – key facts about the enterprise;
- Function – key operation necessary to run the enterprise;
- Platform – technology that enables ‘data’ and ‘function’;
- People – consumers/providers of the ‘data’, users of the ‘functions’, and operators of the ‘platforms’ (Carbone, 2004: 46).

The columns of the framework address the required architectural outputs, *i.e.*:

- Principles – statement of direction as abstracted from the business target state;
- Models – graphical representation of the business view, based on context and ground rules;
- Inventory – list of all key IT resources and their key attributes;
- Standards – corporate set of key standards, including names, definitions and descriptions (Carbone, 2004).

All the different elements are structurally related, as all components are associated with one another.

5.4.2.2 Strategic Technology Architecture Roadmap

Radhakrishnan (2006) patented the 'Strategic Technology Architecture Roadmap' (STAR), which can provide "an end-to-end framework for all or all [sic] or substantially all the technologies in an enterprise, including those used by its trading partners and external customers." STAR encompasses and ties together all the different architectures, like networks, integrations and applications, in an enterprise and towards an enterprise's goals.

Figure 5-7 illustrates this framework, with different embodiments or layers (Radhakrishnan, 2006):

- Upper Infrastructure Layer – service-driven architecture, for example, network management servers, problem management servers (*e.g.* Remedy), web servers (*e.g.* Apache), application servers, middleware, *etc.*
- Virtual Application Layer – distributed component architecture, for example, J2EE (Java 2 Enterprise Edition) technology, servlets, applets, *etc.*
- Application Infrastructure Layer – n-tier architecture, for example, PKI (Public Key Infrastructure), XML, LDAP (Lightweight Directory Access Protocol), SQL (Structured Query Language), HTML (HyperText Markup Language), *etc*.
- Compute Infrastructure Layer – adaptive compute architecture, for example, UltraSPARC (Scalable Processor ARChitecture), Gigaplane, *etc*.
- Network Application Layer – network centric architecture, for example, VLAN (Virtual Local Area Network), TCP/IP (Transmission Control Protocol/Internet Protocol), DMZ, *etc.*
- Storage Infrastructure Layer – storage network architecture for example, NAS (Network Attached Storage), DAS Direct Access Storage, SAN (Storage Area Network), *etc*.

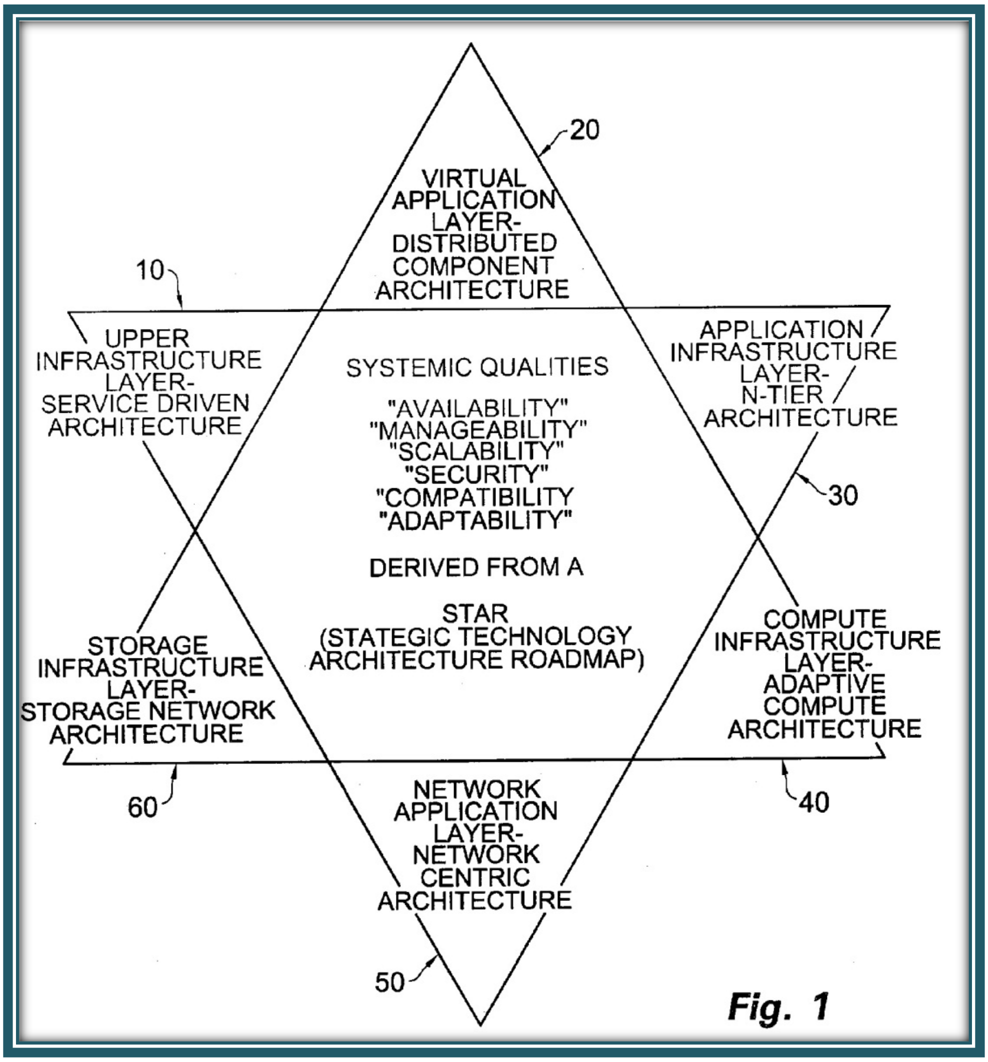


Figure 5-7: Strategic Technology Architecture Roadmap (Radhakrishnan, 2006)

The embodiments described here are not the only possibilities, other embodiments can be substituted or the existing embodiments can be modified, omitted, combined, adapted and/or altered (Radhakrishnan, 2006).

5.4.2.3 Technology Architecture Framework (Victoria University)

The Victoria University of Wellington has documented their 'Technology Architecture Framework' in a series of diagrams. A high-level overview is provided in Figure 5-8.

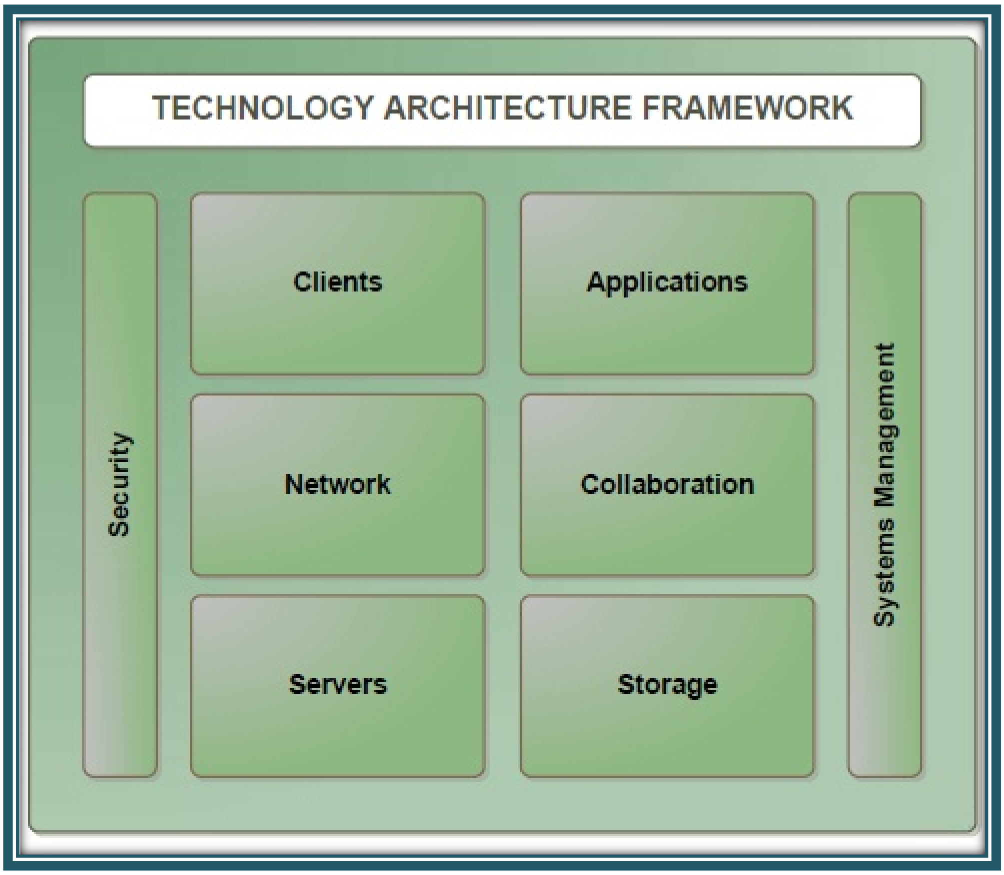


Figure 5-8: Victoria University of Wellington TA (Bates & Nelson, 2009: 1)

The framework consists of six components in three rows:

- Clients – consisting of a roadmap with fixed and flexible infrastructure for staff and students;
- Applications – consisting of applications for the different business units and their specific functionalities as well as the interrelationships or integrations between these functions;
- Network – consisting of the core, LANs (Local Area Network), DR (Disaster Recovery) site, firewall, load balancers, ISP (Internet Service Provider) links and wireless;
- Collaboration – consisting of federated networks, shared services, email, instant messaging, video conferencing, IP (Internet Protocol) phones and presence;
- Servers – consisting of physical and virtual servers and a roadmap for new servers;
- Storage – consisting of central service storage, research storage, archive storage and operational and disaster recovery storage (Bates & Nelson, 2009).

The framework also has two vertical cross-cutting components:

- Security – consisting of several security zones, namely red (core systems), orange (internal servers), yellow (external servers), green-2 (semi-trusted clients) and green (untrusted clients);
- Systems Management – consisting of the monitoring, management, configuration and hardware across all six the components described above (Bates & Nelson, 2009).

5.4.3 Modelling Languages

Models, as reasoned in Chapter 2, are used to fill in the different structures and sections of a framework. The generic models (see paragraph 2.8) like the process model, the relational model, a causal loop diagram and the object-oriented model, can thus also be applied to technology architecture.

There are other models that can also be used to fill in the sections or views of a framework:

- A Data Flow Model – which combines physical and logical diagrams to indicate all the elements of the data flow, for example, middleware (Addison, 2007: 3).
- MEGA Architecture – "ensures the description and consolidation of IT systems maps in a common repository" to facilitate IT asset governance, IT planning and roadmaps and all other EA initiatives (Anon., 2012a).
- CORA (COmmon Reference Architecture) Model – creates a reference software architecture or detailed technology architecture from a total IT landscape point of view as a "basis from which solution architects of individual projects within the portfolio can be derived and detailed". It is vendor agnostic (Elzinga, Smiers & Van der Vlies, 2010).

It is important to utilise a standardised modelling language throughout an enterprise in order to gain alignment, maintain consistent models and enhance efficiency (State of North Carolina. Office of the Chief Information Officer, 2005: 10). There are a variety of descriptive languages, which can be used to express the content of the frameworks or model. A few examples, relevant to technology architecture, are:

- The Unified Modelling Language (UML) – "defines the industry-standard notation and semantics for object-oriented and component-based systems" (Ambler, 2009).
- Web Services Description Language (WSDL) – is an interface description language for describing the interfaces and services and allows web services applications to publish and discover the services, interfaces, methods, protocols, and procedures for communicating between endpoints (State of North Carolina. Office of the Chief Information Officer, 2005: 22). It "describes the operations supplied by services, including expected parameters and return values" (Lucchi & Mazzara, 2007: 97).
- Domain Specific Language (DSL) – "designed to deal with business models in a readable, concise and intuitive way" by making use of layers, where each layer encapsulates functionality of the layers below and enables higher levels of abstraction in the upper layers (Neef, 2011: 10).

5.4.4 Ontologies

Ontologies, as defined in paragraph 3.4.4, are used to provide relationships between elements as well as providing the semantic context. Ontology uses the origins of a concept to classify the concept into a schema and to portray the interrelations. A few of the ontologies in literature, relevant to IT architecture, are summarised below:

- Health Level 7 (HL7) – a well-established message-based standard for data exchange (based on a comprehensive catalogue of message triggering events and the associated message formats) which contributes to reducing semantic heterogeneity (Beyer *et al.*, 2004: 268).
- TDWG Ontology Metamodel (TOM) – The Taxonomic Databases Working Group (TDWG) created an application to manage their specific ontology, containing classes, properties, relationships, hierarchy and single instances (Hyam, 2006).
- Enterprise interaction ontology – to provide "a precise communication scheme for more effective collaboration during architecture evolution" (Kumar *et al.*, 2008: 308).
- Sysperanto – consists of a combination of "generality (covering the IS field), vocabulary (identification of terms), and structure (internally consistent organization)" to create "an organizing framework for codifying the disparate and inconsistent propositions, methods, and findings that constitute the current state of IS knowledge

and, in combination, form a major obstacle to knowledge accumulation and use in the IS field" (Alter, 2005: 1).

5.5 SUMMARY AND CONCLUSION

5.5.1 Summary

The sub-research question answered in this chapter is: *f) What definitions, benefits and models are there for Technology Architecture?*

The question was addressed by first evaluating existing definitions of technology architecture and then compiling a working definition thereof. The synthesised working **definition,** as depicted in Figure 5-1, is duplicated here in Figure 5-9 for ease of reference and is written out as:

> *Technology architecture is the process of describing, modelling, communicating, applying, governing and maintaining the current state, future-state blueprints, interrelationships and change/innovation of the IT systems, infrastructure, strategy, portfolio and investment of an enterprise.*

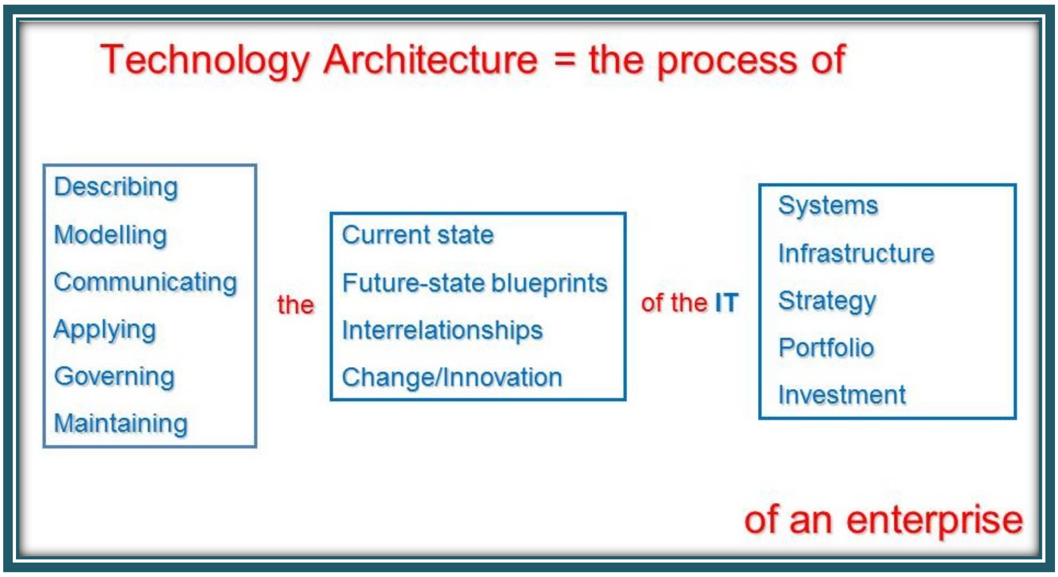


Figure 5-9: Synthesised Definition of Technology Architecture

Second, facets of technology architecture's **role** within the enterprise are, in short, to govern IT investments and assets, describing the IT plans, infrastructure and interrelations, and enabling innovation and flexibility. Technology architecture can, in a nutshell, produce **benefits** by reducing cost, improving decision-making, operational effectiveness and business alignment and providing growth in flexibility.

Third, the different levels of abstraction with regard to **frameworks** and **models** were discussed. Table 5-2 provides a quick overview of the abstraction levels.

TYPES	*EXAMPLES*
EA Frameworks	Technology, Components & Operations (Zachman Framework) Technology Architecture (TOGAF) Technical Standards View (DoDAF) Technical Reference Model (FEAF) Gartner IT Architecture Guideline Framework
TA Frameworks	IT Framework (Carbone) Strategic Technology Architecture Roadmap Technology Architecture Framework (Victoria University)
Models	Data Flow Model MEGA Architecture CORA Model

Table 5-2: Overview of Levels of Frameworks and Models (Technology Architecture)

5.5.2 Conclusion

A synthesised definition of technology architecture has been provided. Technology architecture has specific roles within an enterprise and can provide significant benefits to the enterprise. There are different frameworks, models, ontologies and descriptive languages, which can be utilised to describe an enterprise's technology architecture and create architecture artefacts.

Technology architecture is an important part of enterprise architecture as technology architecture is the realisation of solutions for business requirements via IT infrastructure and systems through IT planning and governing. Technology architecture thus completes the cycle for aligned business requirement fulfilment.

Technology architecture will play a significant role as enabler in the creation of an overarching architectural model, interlinking the different architectural domains. The next chapter (Chapter 6) will describe the interlinking between the different domains of enterprise architecture.

6 RELATIONSHIPS BETWEEN THE ARCHITECTURAL DOMAINS

6.1 INTRODUCTION

In the previous chapters the different concepts and domains encompassing enterprise architecture were described. The crux of successful enterprise architecture modelling lies in the 'glue' which interlinks these architectural domains on several levels. *Glue* originates from the Latin word *glutinium.* The Greek word for glue is *kolla* and is still seen today in words such as *collagen*. To glue means to attach, to join, to bind or to cement. These words are also recognisable in the Spanish words *cemento* and *aglutinante*. Glue is the *adhesive* necessary to provide *cohesion* between the architectural domains. Choosing the best term to use for the 'glue' is discussed in paragraph 6.3.

Gartner (Guevara & Robertson, 2011: 1) found that in general EA guidance fails to be consistently applied by solution delivery teams, which results in missing critical dependencies and in inconsistent delivery of strategy. "The use of EA frameworks to 'divide and conquer' the complexity of EA analysis seems to work until these viewpoints must reunite as guidance to solution delivery teams." The coherence or 'glue' is thus still a shortfall.

Different mechanisms to achieve the interlinking or cohesion will be discussed of which one is language. According to Lankhorst *et al.* (2005: 83) "the current situation is that architects in different domains, even within the same enterprise, often use their own description techniques and conventions" and that no standard language exists for describing enterprise architectures precisely across the different domain borders. They also express the opinion that a separate enterprise modelling language, in addition to the more detailed domain-specific languages, will add value.

Enterprise architecture (especially in large enterprises) has a large scope to cover and the architectural work will have to be distributed to a population of domain architects within the enterprise (Bruls *et al.*, 2010: 518). The planning and control by enterprise architecture should be reflected in all the architectural domains and should be linked to each other. Enterprise architecture is, by definition, an instrument to address enterprise wide integration/interlinking of the different architectural domains (De Vries, 2010: 17; Gerber, Kotzé & Van der Merwe, 2010: 54; Kang, Lee & Kim, 2010: 3274; O'Rourke, Fishman & Selkow, 2003: 7). Schekkerman (2004b: 13) confirms this and sees enterprise architecture as "a complete expression of the enterprise; a master plan which 'acts as a collaboration force' between" all the relevant aspects from the business, information and technology. Figure 6-1 depicts the alignment or interactive nature of enterprise architecture.

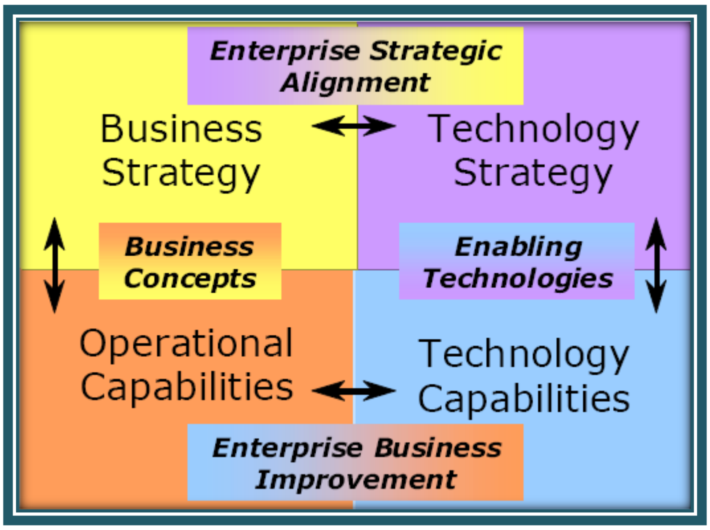


Figure 6-1: The Holistic View of EA (Schekkerman, 2004b: 13)

"However, in practice, these domains are not approached in an integrated way. Every domain speaks its own language, draws its own models, and uses its own techniques and tools" (Lankhorst, 2004: 205). This is confirmed by Saat *et al.* (2010: 14): "However, existing approaches do not distinguish between different IT/business alignment situations." This leads to the next two sub-research questions, namely:

g) *What are the taxonomy and relationships of the different architectural domains?*

h) *What definitions, benefits and models are there for integrating, interacting and/or interlinking the architectural domains or are utilised for indicating the relationships between the architectural domains?*

In response first, the issue of taxonomy and relationships of the different architectural domains will be addressed by an overview of the literature, which leads to a proposed taxonomy and a proposed integrated depiction of EA.

Second, the current status of the integration and interlinking between the architectural domains is described by means of a critical analysis of existing literature, through exploring its definitions as well as its role and benefits. Integration as incorporated by some of the existing frameworks and models is also described briefly. This will provide essential background to create an understanding of the environment and will be used as input in creating a model to facilitate the integration and interlinking between the different architectural layers or domains.

6.2 DEPICTION AND TAXONOMY OF THE ARCHITECTURAL DOMAINS

The different domains encompassed by enterprise architecture will be discussed from the literature. According to general dictionaries a domain is "an area of knowledge" (Webster's New World Dictionary, 2011) or "a specified sphere of activity or knowledge" (Oxford Dictionaries, 2011). In this context 'domains' do not indicate different business domains or functions within an enterprise, but the different layers or domains within enterprise architecture. Different stakeholders observe the enterprise and the architecture from different viewpoints, influenced by their domain of experience (Lankhorst *et al.*, 2005: 53). Bernard (2005: 37) labels these as 'sub-architectures' or 'levels', with distinct functional areas and relationships. O'Rourke (2003: 468) defines a 'domain' as "an area of interest with well-defined boundaries. A domain may contain other domains." The enterprise is the highest level of domain in this context. Lankhorst *et al.* (2005: 87) talk about 'layers', containing internal services, which typically link to other layers by being supported by a layer below or by supporting realisation of a layer above. The term 'architectural domain' will consistently be used throughout this research.

There are quite a number of possible architectural domains which can form part of enterprise architecture. Ross, Weill & Robertson (2006: 48) list four architectural domains: "business process architecture (the activities or tasks composing major business processes identified by the business process owners); data or information architecture (shared data definitions); applications architecture (individual applications and their interfaces); and technology architecture (infrastructure services and the technology standards they are built on).” Sessions (2007: 2) uses the terms ‘business architectural level’, ‘application architectural level’ and ‘IT architectural level’. Jonkers *et al.* (2006: 64) address information, application, process, product and technical architecture as well as their interrelationships. The domains of business architecture (Chapter 3), information architecture (Chapter 4) and technology architecture (Chapter 5) have already been discussed in more detail.

Note: The same colour convention, as described in Chapter 1, will be used, *i.e.* pink will indicate integration/interlinking and pink lines have been added to some of the diagrams for emphasis.

6.2.1 Depiction of Enterprise Architecture

It is important for an enterprise to capture its understanding of enterprise architecture in a simple one-page core diagram to assist managers and executives to understand their enterprise’s enterprise architecture (Ross, Weill & Robertson, 2006: 50). Depictions are also an effective way to indicate the relations between the different architectural domains. Examples of such diagrams of enterprise architecture, as found in the literature, are discussed below.

Figure 6-2 shows Gartner’s (Guevara & Robertson, 2011: 2) representation of EA, by depicting the business, information and technology architecture within the enterprise context. The combination or intersection of the three viewpoints results in solution architecture, with greater consistency, interoperability and portability in solutions. This diagram succeeds in providing a clear depiction of the interrelation between the different domains. The encompassing enterprise context drives the context, change and business requirements of EA.

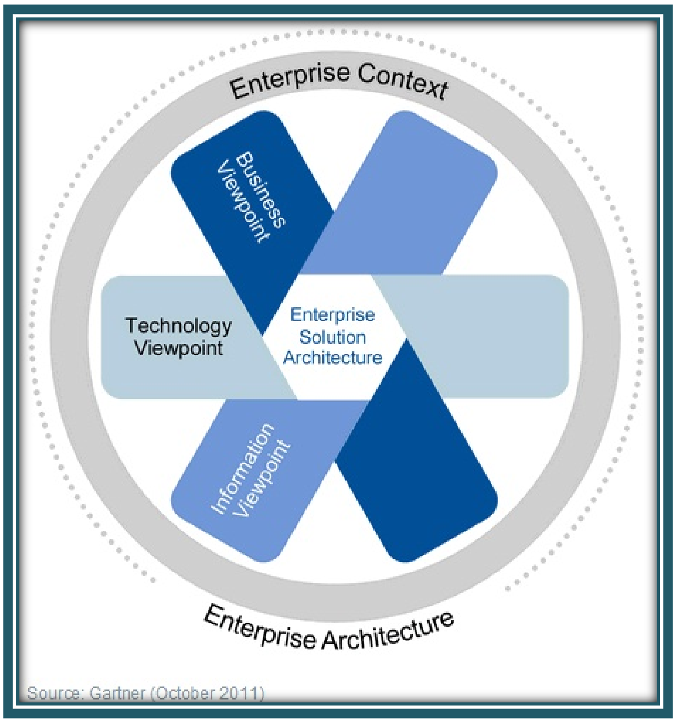


Figure 6-2: Gartner Enterprise Architecture Framework (Guevara & Robertson, 2011: 2)

Figure 6-3 shows Sessions's (2007) representation of EA, by depicting enterprise architecture as the overarching architecture, with business architecture (including processes) and information technology (including applications) as domains. The diagram oversimplifies the concept – there is a trade-off between simplicity/clarity versus comprehensiveness. The main issues are the absence of any reference to information architecture as well as the absence of any reference to the interrelation between IT and business.

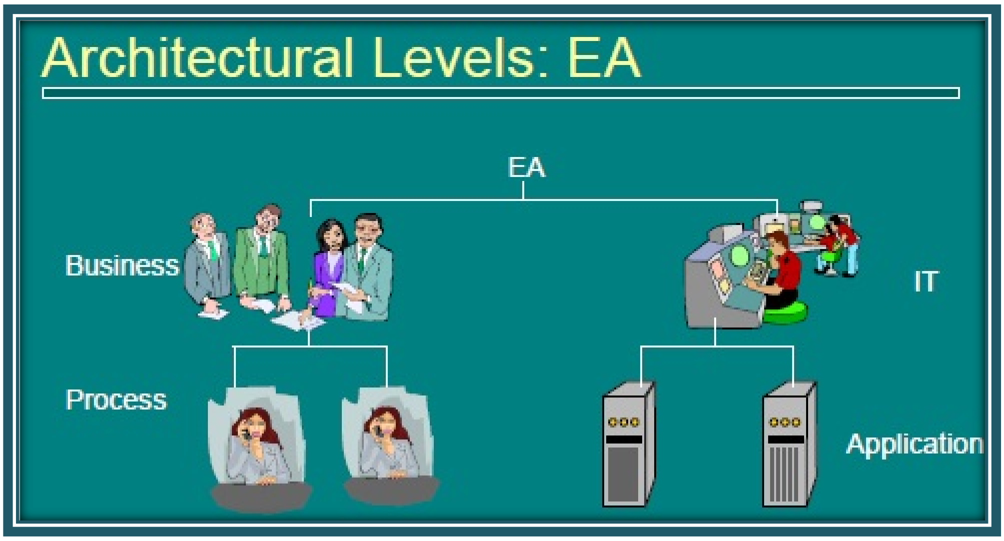


Figure 6-3: Sessions's Representation of Enterprise Architecture (2007)

Figure 6-4 shows the representation of Jonkers *et al.* (2006), by depicting the interrelated architectural domains, *i.e.* information architecture, product architecture, process architecture, application architecture and technical architecture. Although it attempts to indicate the relationships between the different domains through arrows, there is no indication or reference to business architecture (or solutions architecture).

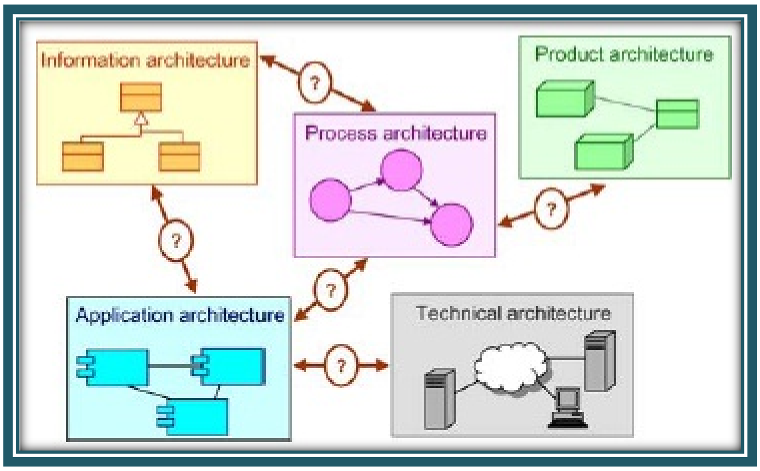


Figure 6-4: Jonkers's Representation of Enterprise Architecture (Jonkers *et al.*, 2006)

Figure 6-5 shows Wu's (2007) representation of EA, by depicting the different architectural domains (business, data, application, technology and security) cross-cutting through stovepipe systems and identifying commonalities. Although it creates the impression of an integrated model there are no connections or cross sectioning of the different architectural domains. Should the order of the layers be significant, it is debatable if data architecture should fit in between application architecture and business architecture. The logical flow is: the business requires an application, which will transact on data, which are hosted on technology.

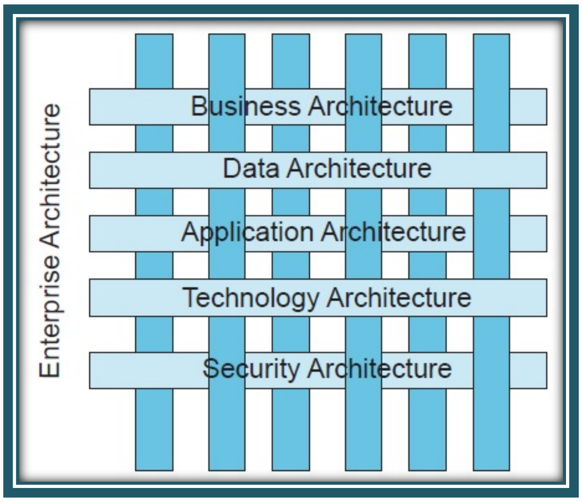


Figure 6-5: Wu's Representation of Enterprise Architecture from www.e-cio.org (Wu, 2007)

Figure 6-6 shows the representation of EA by Alonso *et al.* (2010), by depicting business architecture and IT architecture as the two domains of EA, with governance as a vertical pillar. They include information architecture, data architecture, applications architecture and technology architecture as components of IT architecture. In this depiction IT architecture is heavily loaded with a variety of architectural domains, which may result in too big a focus on IT architecture to the detriment of the business architecture and the enterprise as a whole. A possible solution could be to cluster information architecture and applications architecture into a middle layer.

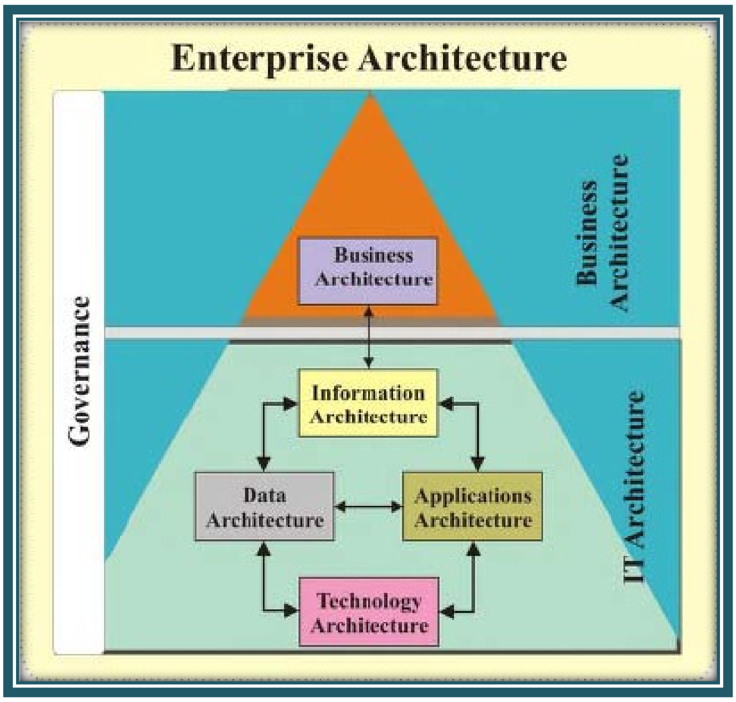


Figure 6-6: Alonso's Representation of Enterprise Architecture (Alonso et al., 2010)

Figure 6-7 shows the representation of Minoli (2008: 17), by distinguishing between logical resources, containing the business services, the information and the (systems) solution layers, and the physical resources, containing the technology layer. Across these layers is the operational/management layer, containing *inter alia* the security and performance functions. In this depiction the level of detail for the technology layer differs from the level of detail for the other layers, the term 'layer' creates the impression of a hierarchy, and the relations between the layers are not a true reflection of Minoli's point of view.

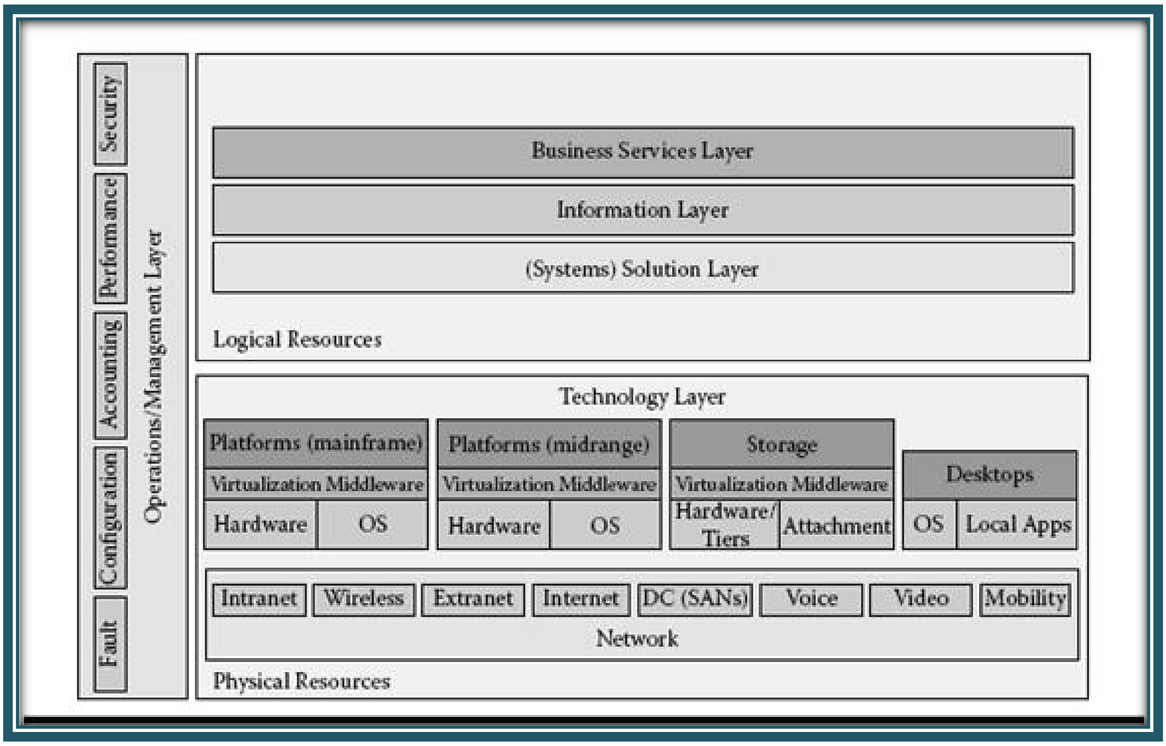


Figure 6-7: Minoli's Layered Model of Enterprise Architecture (2008: 17)

It is thus evident from these examples that there is a variety of different depictions of enterprise architecture.

6.2.2 <u>Taxonomy of Enterprise Architecture</u>

The variety of EA definitions discussed in Chapter 2 (see 2.2.1) and the six different diagrams in 6.2.1 indicate that there is a variety of descriptions and depictions of EA, with some similarities but limited consensus. In order to present an integrated diagram of enterprise architecture, a schema or taxonomy of all the architectural domains was compiled. There is a variety of terms being used, some as synonyms and other on different levels of architecture. The terms were grouped and organised to provide a tree-like proposed taxonomy of enterprise architecture, see Table 6-1.

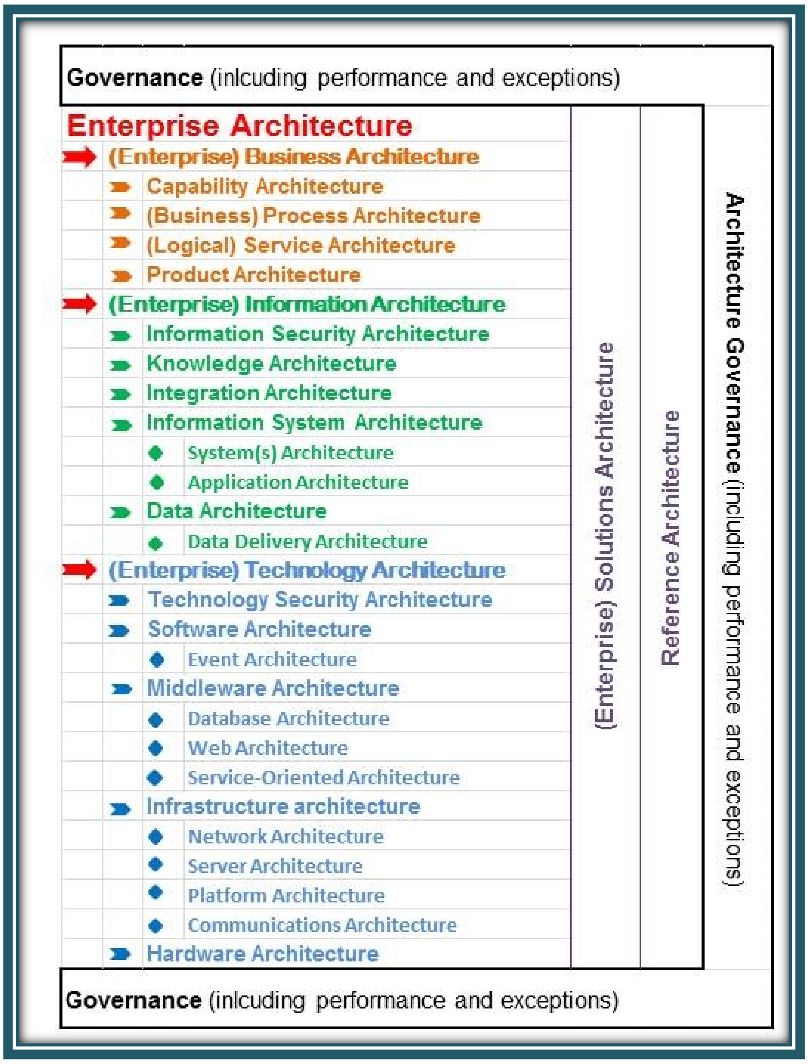


Table 6-1: Proposed Taxonomy for Enterprise Architecture

A selection was made from the different possible components of EA as found in the literature to provide an illustrative taxonomy, albeit not necessarily comprehensive:

- It contains three main domains, *i.e.* business architecture, information architecture and technology architecture.
- Beneath each domain a set of sub-domains is proposed, which are not automatically all on the same level across the different domains.
- Some of the sub-domains are sub-divided further with another layer of architectural components.
- Solutions architecture is proposed as a cross-cutting architectural domain, which aggregates elements from all the other domains in order to design solutions for the enterprise.
- Reference architecture is also depicted as cross-cutting, because every component could have its own reference architecture

- The governance of the enterprise architecture process is portrayed as a frame managing the whole enterprise architecture domain and all components within.

6.2.3 A Proposed Integrated Depiction of Enterprise Architecture

As a summation of the above taxonomy (Table 6-1), a possible integrated representation of the architectural domains is displayed in Figure 6-8. The diagram depicts enterprise architecture, as the focus and integrated result, in the middle. Solutions architecture is seen as a cross-cutting result of the other three main domains. The three domains of business, information and technology architecture all have connections with and influence on each other. The chosen combination of artefacts is described in the reference architecture and shown as an outside circle to support the architecture process. Different perspectives or views are not addressed *per se* in this diagram.

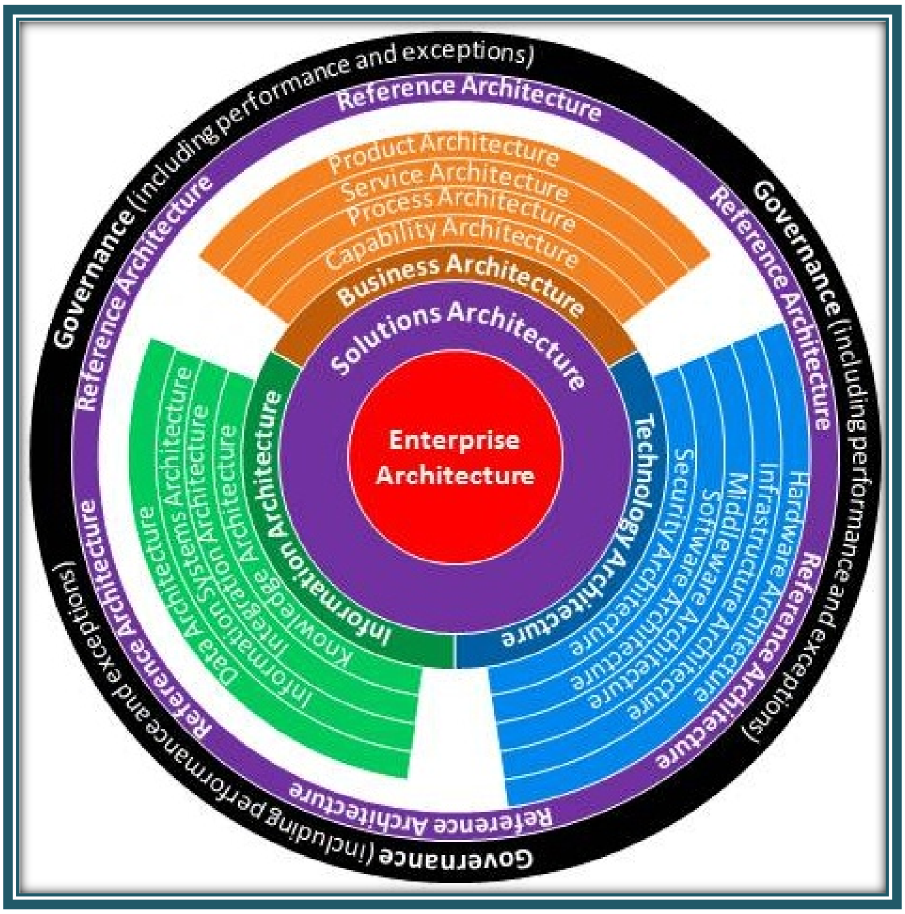


Figure 6-8: The Proposed Integrated Diagram of EA

This single diagram adds value as it provides a holistic view on EA, its main domains and its possible sub-domains in one depiction. This is essential for understanding, promoting and conveying EA within the enterprise and executive. It also attempts to express the interrelations and alignment between the main domains, which aggregate in solutions architecture. Reference architecture is depicted as an outside circle to indicate its relevance to all the architectural domains and sub-domains. Governance once again forms a frame, as it manages the whole EA process within the enterprise.

6.3 DEFINING THE RELATION BETWEEN THE ARCHITECTURAL DOMAINS

It seems problematic to obtain literature providing a definition of the inter-relationships and integration of the different architectures. This may be the result of the interdisciplinary nature of the topic and/or the lack of a definitive phrase/label to identify this concept. For example, Finkelstein (2006: 15) "discussed that enterprise integration depends on business integration and also technology integration. Business integration is achieved through the use of enterprise architecture and related enterprise engineering methods. Technology integration is achieved with the use of XML, EAI (Enterprise Application Integration), enterprise portals, Web services, and SOA."

Vernadat (2007: 138) gave the following description: "Enterprise integration occurs when there is a need in improving interactions among people, systems, departments, services, and companies". The term 'integration' is often found in EA literature. It does, however, signify different meanings and levels, for example, integration of:

- different software products;
- a range of technologies;
- EA in separate business units (see Figure 6-9 as an example);
- the enterprise's partners and customers;
- the supply chain;
- the business strategy and technology implementations; or
- database schemas.

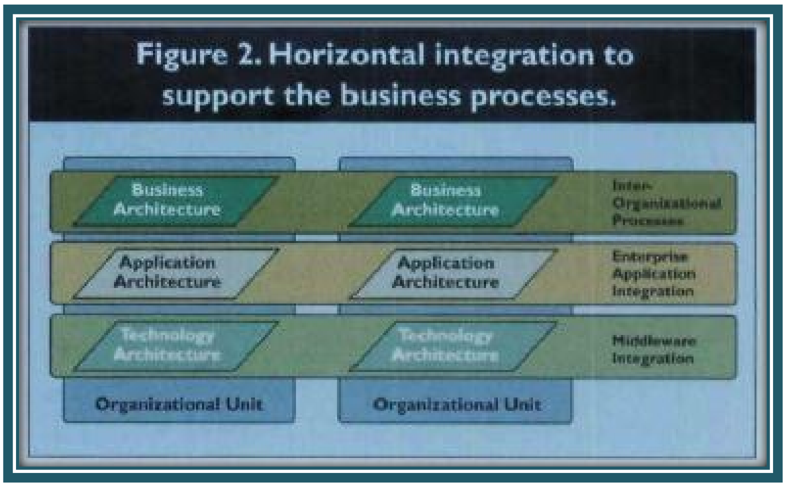


Figure 6-9: An Example of Integration of Unit EAs (Hasselbring, 2000: 35)

Aier & Schönherr (2006) state that: "The current discussion about integrative enterprise architectures is not very structured" "and industry standard is far away". In the absence of proper existing definitions, the different terms or descriptions that are used in the literature to describe this concept have been investigated. An alphabetical list of some of these terms has been compiled, and each term is followed by a dictionary definition:

- Addressing a level in-between the different architectures (Bruls *et al.*, 2010: 518). In-between: "between two definite or accepted stages or states, and therefore difficult to describe or know exactly" (Cambridge Dictionaries Online, 2011).
- Alignment (Bruls *et al.*, 2010: 521; Finkelstein, 2006: 243; Gerber, Kotzé & Van der Merwe, 2010: 17; Kang, Lee & Kim, 2010: 3274; Lankhorst *et al.*, 2005: 221; O'Rourke, Fishman & Selkow, 2003: 7; Pessi, Magoulas & Hugoson, 2011: 53; Saat *et al.*, 2010: 14). Alignment: "when two or more things are positioned in a straight line or parallel to each other" (Cambridge Dictionaries Online, 2011).
- Architecture Management/Control (Aier & Schönherr, 2006; Bruls *et al.*, 2010: 518). Management: "the control and organization of something" (Cambridge Dictionaries Online, 2011).
- Bridging the gaps (De Vries, 2010: 24; 2012: 135). Bridge a/the gap: "to connect two things or to make the differences between them smaller" (Cambridge Dictionaries Online, 2011).
- Central plexus (Whittle & Myrick, 2005: xv). Central: "equally distant or accessible from various points" (Webster's New World Dictionary, 2011). Plexus: "a complexly interconnected arrangement of parts" (Webster's New World Dictionary, 2011).

- Coherence (Lankhorst *et al.*, 2005: 147; Saha, 2007: 7). Coherence: "when the parts of something fit together in a natural or sensible way" (Cambridge Dictionaries Online, 2011).
- Continuity (O'Rourke, Fishman & Selkow, 2003: 573). Continuity: "the state or quality of being continuous; connectedness; coherence" (Webster's New World Dictionary, 2011).
- Cross-cutting (Wu, 2007: 1). Cross-cutting: "a technique used especially in filmmaking in which shots of two or more separate, usually concurrent scenes are interwoven" (Webster's New World Dictionary, 2011).
- Integration (Aier & Schönherr, 2006; Finkelstein, 2006: 15; Huysmans, Ven & Verelst, 2010: 77; Lankhorst *et al.*, 2005: 83; Minoli, 2008: 17; O'Rourke, Fishman & Selkow, 2003: 7; Rohloff, 2008: 561; Vernadat, 2007: 137). Integration: "to combine or join two things in order to become more effective" (Cambridge Dictionaries Online, 2011). Integration: " mixing things or people together that were formerly separated" (Webster's New World Dictionary, 2011). Integration: "combining parts so that they work together" (WhatIs.com, 2012).
- Interaction (Rohloff, 2008: 564). Interaction: "when two or more people or things communicate with or react to each other" (Cambridge Dictionaries Online, 2011).
- Interdependencies (Rohloff, 2008: 561; SOA Consortium EA2010. Working Group, 2010: 1). Interdependent: "depending on each other" (Cambridge Dictionaries Online, 2011).
- Inter-related models (SOA Consortium EA2010. Working Group, 2010: 4). Interrelated: "to be connected in such a way that each thing has an effect on or depends on the other" (Cambridge Dictionaries Online, 2011).
- Link/linking/linkage (Lankhorst *et al.*, 2005: 86; Saha, 2007: 7; SOA Consortium EA2010. Working Group, 2010: 1). Linking: "connects the qualities of an object or person to that object or person" (Cambridge Dictionaries Online, 2011).
- Mapping (Minoli, 2008: 9). Mapping: "a transformation taking the points of one space into the points of the same or another space" (Webster's New World Dictionary, 2011).
- Relationship or relations (Aier *et al.*, 2009: 36; Kang, Lee & Kim, 2010: 3274; Lankhorst *et al.*, 2005: 83; Minoli, 2008: 17; Pessi, Magoulas & Hugoson, 2011: 53;

SOA Consortium EA2010. Working Group, 2010: 1). Relationship: “the way in which two things are connected” (Cambridge Dictionaries Online, 2011).

These terms are not all indicating the same level of integration. According to The Open Group (2009c: 63) there “are varying degrees of architecture description ‘integratability’”. The low end is where integratability indicates a sufficiently similar “look-and-feel” between different architecture descriptions to identify critical relationships between the descriptions or to at least indicate the need for further investigation. The high end is where integratability ideally indicates that different descriptions should be capable of being combined into a single logical and physical representation.

The above terms were thus placed on a relationship continuum between independent components and a fully integrated whole (labelled as the relationship scale) – see Figure 6-10. The term ‘interlinking’ was added: “to cause to join or connect together, with the parts joined often having an effect on each other” (Cambridge Dictionaries Online, 2011). The term was added, because:

- ‘inter-‘ (the Latin for between, among, within or connecting) appears as a prefix for three of the terms used in the literature;
- ‘link’ appears three times in the referenced literature; and
- the combination of the two terms adds value to the meaning of the concept.

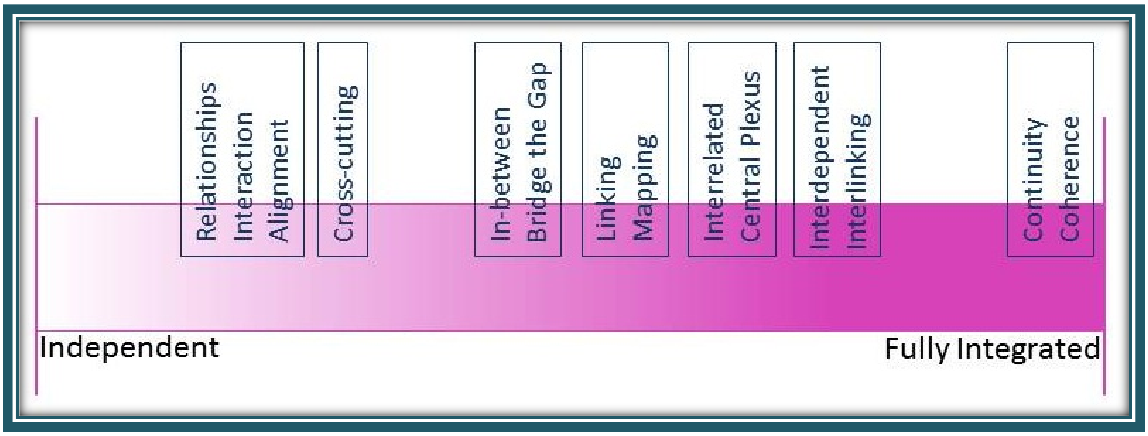


Figure 6-10: Relationship Scale of Terms

The frequency of the terms is illustrated in Figure 6-11. It is clear from this graph of the different terms being used in the literature, that the terms ‘alignment’, ‘integration’ and ‘relations’ have the highest occurrence. Based on the analysis of these terms

- the terms '*relation' / 'relating'* will be used throughout the rest of this thesis to indicate the overall concept of combining different architectural domains;
- the term *'integration strategy'* will be used throughout the rest of this thesis to indicate the scope or components (abstraction layers, perspectives, viewpoints, (De Vries, 2010: 19)) to be included in the combined architecture – the 'what'; and
- the term *'interlinking mechanism'* will be used throughout the rest of this thesis to indicate the mechanisms (processes, methodologies, tools and governance structures (De Vries, 2010: 19)) used to provide the cohesion between the architectural domains – the 'how' or the 'glue'.

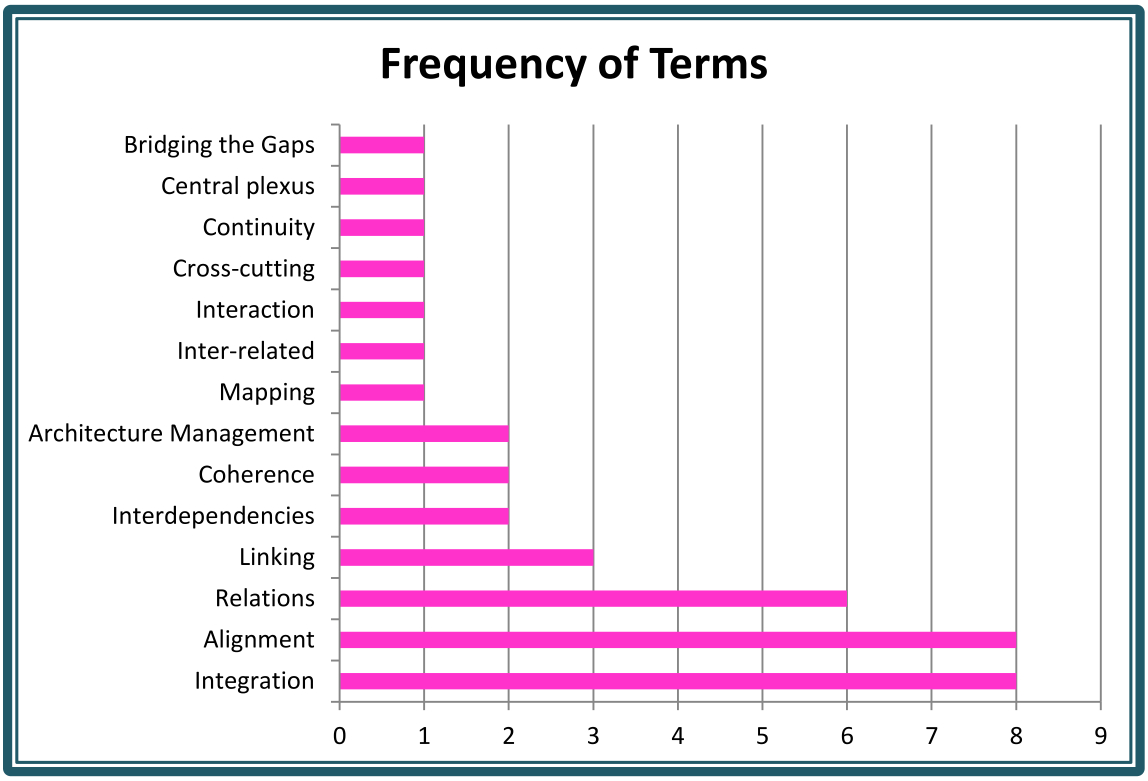


Figure 6-11: Frequency of Terms in the Literature

A definition was subsequently developed and is depicted in Figure 6-12: Synthesised Definition of Relating of Architectural Domains. The full definition is:

> *Relating architectural domains is the integration strategies and the interlinking mechanisms used to combine two or more of business architecture, information architecture and technology architecture to provide meaningfully aligned enterprise architecture, in the context of an enterprise.*

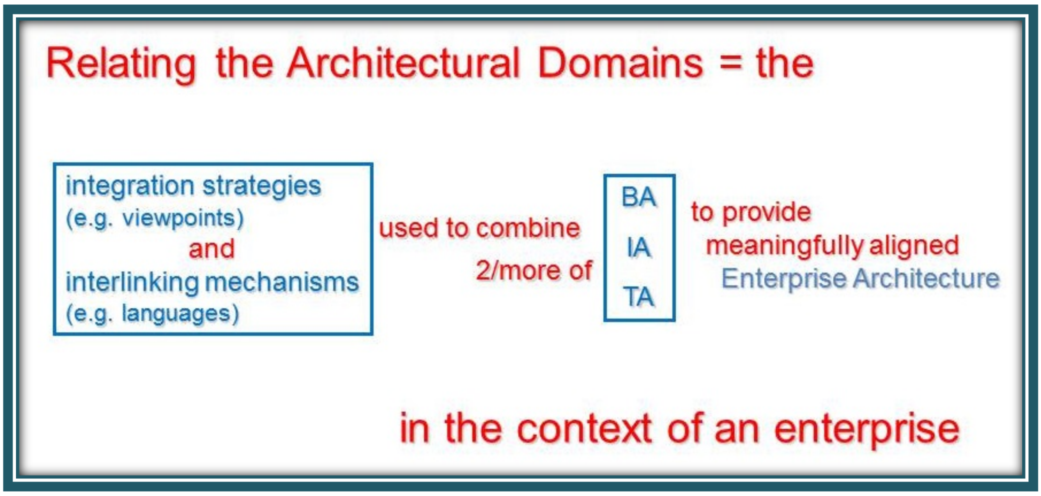


Figure 6-12: Synthesised Definition of Relating of Architectural Domains

6.4 THE ROLE AND BENEFITS OF RELATING THE ARCHITECTURAL DOMAINS

Defining the relationships between the different architectural domains has a role to fulfil within the enterprise architecture stack as well as specific contributions to successful enterprise architecture.

Some of the facets of this **role**, as found in the literature, are:

- Enabling co-ordinated investments through the enterprise (Pessi, Magoulas & Hugoson, 2011: 53).

- Providing “a clear view on the structure of and dependencies between relevant parts of the organization” (Saat *et al.*, 2010: 18).

- Modelling any “global structure *within* each domain, showing the main elements and their dependencies”;
- modelling the relevant relations between the domains; and
- ensuring models which are unambiguous and amenable to automated analysis (Lankhorst *et al.*, 2005: 84).

- Providing a common language through the different architectural components and enterprise (Lankhorst *et al.*, 2005: 83; Saat *et al.*, 2010: 15; Whittle & Myrick, 2005: 60).

- Managing of complexities (Pascot, Bouslama & Mellouli, 2011: 116).

- Enabling innovation (Huysmans, Ven & Verelst, 2010: 76).

- Removing discontinuity across the enterprise (O'Rourke, Fishman & Selkow, 2003: 573).

Some of the **benefits** of defining the relationships between the different architectural domains, from the literature, are:

- Insight into the relationship between EA and IT-investments (Pessi, Magoulas & Hugoson, 2011: 53) and the value from IT investments for the enterprise (Henderson & Venkatrman, 1999: 472).
- Changes and innovation in business processes (Aier & Schönherr, 2006; Lapkin *et al.*, 2008: 4).
- Enabled business processes (Aier & Schönherr, 2006).
- The re-use and sharing of, for example, language, processes, templates (Saat *et al.*, 2010: 18; The Open Group, 2009a; Wu, 2007: 2).
- Conceptually normalised models, where discontinuity or conceptual redundancy has been eliminated (O'Rourke, Fishman & Selkow, 2003: 573).
- Accelerated integration of important applications that require interchange (Object Management Group, 2011: 5).
- Clear communication (Morar, 2008: 22).
- An understanding of aligned/integrated business and IT goals (Burton, 2009: 1).
- The capability to leverage technology to differentiate the enterprise from its competitors (Henderson & Venkatrman, 1999: 473).

Relating the architectural domains thus has an important role within an enterprise for clear, global, co-ordinated, unambiguous modelling to manage complexities and enable innovation. Relating the architectural domains can also produce benefits by enabling the innovation/change and business processes, flexibility and the re-use/sharing of a variety of objects.

6.5 EXISTING FRAMEWORKS AND MODELS

Relating the architectural domains is dependent on the different architectural frameworks. First, the relationship aspects of the main enterprise architecture frameworks are discussed. Second, a number of frameworks, which illustrate these relationships, are described briefly. Third, a number of relevant models/methods are listed.

6.5.1 The Relational Properties of Enterprise Architecture Frameworks

The enterprise architecture frameworks, discussed in paragraph 2.7, have specific integration strategies and interlinking mechanisms to define the relationships between the different architectural domains. These will be discussed below.

6.5.1.1 Zachman Framework

The Zachman framework is organised as 36 cells arranged in a six-by-six two-dimensional matrix. Every cell is by definition an intersection between a perspective and an aspect. There are also relationships between columns. “The Framework further implies the need for enterprise wide integration – achieving continuity across the scope of the enterprise within any and every cell” (O'Rourke, Fishman & Selkow, 2003: 573).

The main purpose of the matrix framework is to bridge the gap between business people and IT people in communicating effectively. De Vries (2010: 24) found that the Zachman framework uses perspectives and aspects to ensure that all requirements are addressed. It is primarily used to facilitate “continuous alignment of business requirements with information system functionality and its supporting infrastructure”.

The intersections are between a column and a row. Finkelstein (2006: 245) adds to this by listing key strategic alignment matrices to integrate between columns:

- Column 6 (why/future) to column 4 (who/people) – indicating the people responsible for the key planning statements. This shows which organisational units are involved in each planning statement and reinforce the unit’s reason for existence.
- Column 6 (why/future) to column 1 (what/data) – indicating the data supporting the key planning statements. This shows what data are needed and why it is needed.
- Column 2 (how/function) to column 1 (what/data) – indicating the data required by key business activities. This shows how activities are used and what data are required.

* Column 6 (why/future) to column 2 (how/function) – indicating activities to support key planning statements. This shows how activities are used and why they exist.

O'Rourke, Fishman & Selkow (2003: 586) created a third dimension to the Zachman framework called Zachman DNA or Zachman Depth iNtegrating Architecture. They made use of the theory of fractals, chaos, DNA (deoxyribonucleic acid) and SDLC (System Development Life Cycle). Where the two dimensions address the types of artefacts and descriptive representations, the third dimension addresses an infinite number of 'sciences' (broadly based on SDLC) or processes. These represent integrated activities to produce the descriptive artefacts. Each science results in a distinct set of descriptive artefacts for the framework. Examples of sciences are:

* project management;
* project administration;
* testing;
* methodology;
* principles;
* standards;
* stakeholders' involvement;
* user involvement;
* change control; and
* version control (O'Rourke, Fishman & Selkow, 2003: 608).

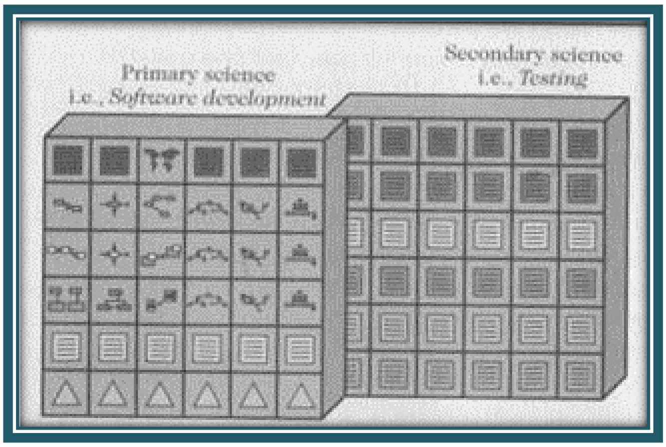


Figure 6-13: Zachman DNA Example (O'Rourke, Fishman & Selkow, 2003: 615)

Based on the fact that everything in Zachman DNA has an *x*, *y* and *z*-axis, everything has contextual meaning. Each 'science' is thus like a chromosome strand with 36 genes. "Pictorially, the first science listed is the primary science. In other words the first science is the focus of the project team", with the secondary sciences supporting the primary science for a specific project (O'Rourke, Fishman & Selkow, 2003: 615). One example is illustrated in Figure 6-13. As with fractals, every cell can again be divided into 36 cells indicating specific focuses like scope and detail; drag, weight, thrust and lift or time and cost. The Zachman DNA can thus be utilised as a framework to view the enterprise as a whole, and on different levels down to microscopic level. The interaction is illustrated in Figure 6-14.

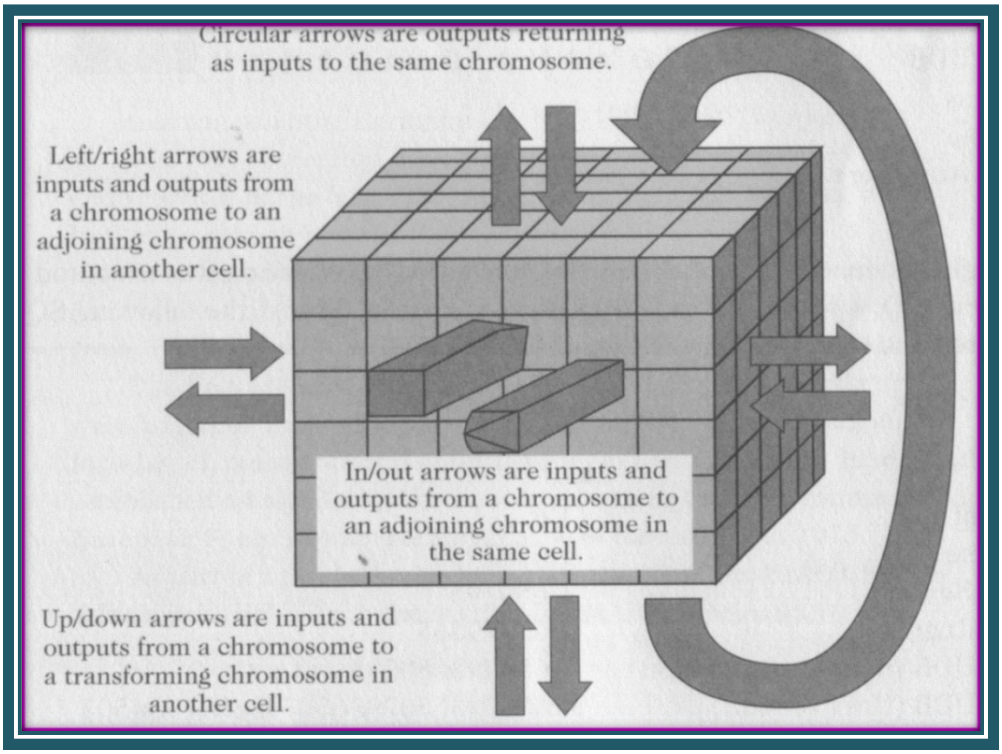


Figure 6-14: Zachman DNA Interaction (O'Rourke, Fishman & Selkow, 2003: 625)

Graves (2010) also did enhancement on the Zachman Framework by adding dimensions, as depicted in Figure 6-15. These are called *segments* or sub-categories within the columns, and will typically be:

- "**physical**: tangible objects (What), mechanical processes (How), physical locations (Where), physical events (When); also align to ***rule-based*** skills (Who) and decisions (Why)

- **virtual**: intangible objects such as data (What), software processes (How), logical locations (Where), data-driven events (When); also align to ***analytic*** skills (Who) and decisions (Why)
- **relational**: links to people (What), manual processes (How), social/relational locations (Where), human events (When); also align to ***heuristic*** skills (Who) and decisions (Why)
- **aspirational**: principles and values (What), value-webs and dependencies (Where), business-rules (When); also align with ***principle-based*** skills (Who) and decisions (Why)
- **abstract**: additional uncategorised segments such as financial (What, How), time (When)" (Graves, 2007).

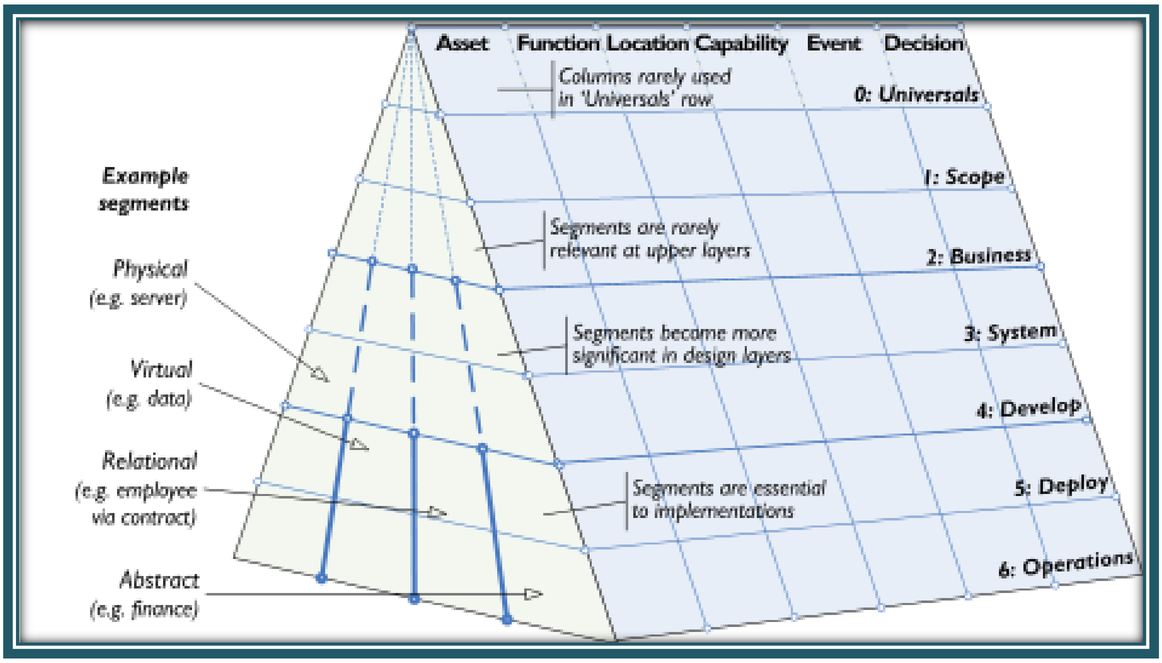


Figure 6-15: Revised Zachman Framework (Graves, 2010)

The pure Zachman framework thus has some integration built in through the intersection of the building blocks in the columns and rows. The interlinking mechanisms are, however, dependent on the individual(s) documenting the EA and their insight. This can be expanded by also matching specific columns with each other (Finkelstein, 2006: 245). The Zachman DNA uses the same building blocks but adds a number of third dimensions of interlinking mechanisms, called 'sciences' (O'Rourke, Fishman & Selkow, 2003: 586). Graves (2010) also adds a third dimension, called segments, containing *inter alia* processes.

6.5.1.2 TOGAF

The basic principle behind TOGAF is to provide a metamodel that encompasses all the different architectural domains consistently (see 6.5.2.4 for description of the term metamodel). This should result in "an integrated environment that is responsive to change and supportive of the delivery of the business strategy" (The Open Group, 2009a). Metamodels are a way of integrating different models and ensure interoperability (Gerber, Kotzé & Van der Merwe, 2010: 54).

TOGAF makes use of the so-called 'Integrated Information Infrastructure Reference Model' (part of the foundation architecture), which is aimed at developing the vision of boundaryless information flow (Minoli, 2008: 85). De Vries (2010: 24) states that the iterative ADM (including a 'Requirements Management') would ensure continuous alignment between the different architectural domains.

TOGAF's metamodel consists of five major parts (The Open Group, 2009a):

- Architecture Development Method (ADM) – a method for developing and managing the life-cycle of enterprise architecture to integrate all the architectural elements and assets to meet the business and IT needs of an enterprise.
- Architecture Content Framework – "a structural model for architectural content that allows the major work products that an architect creates to be consistently defined, structured, and presented" – see Figure 6-16.

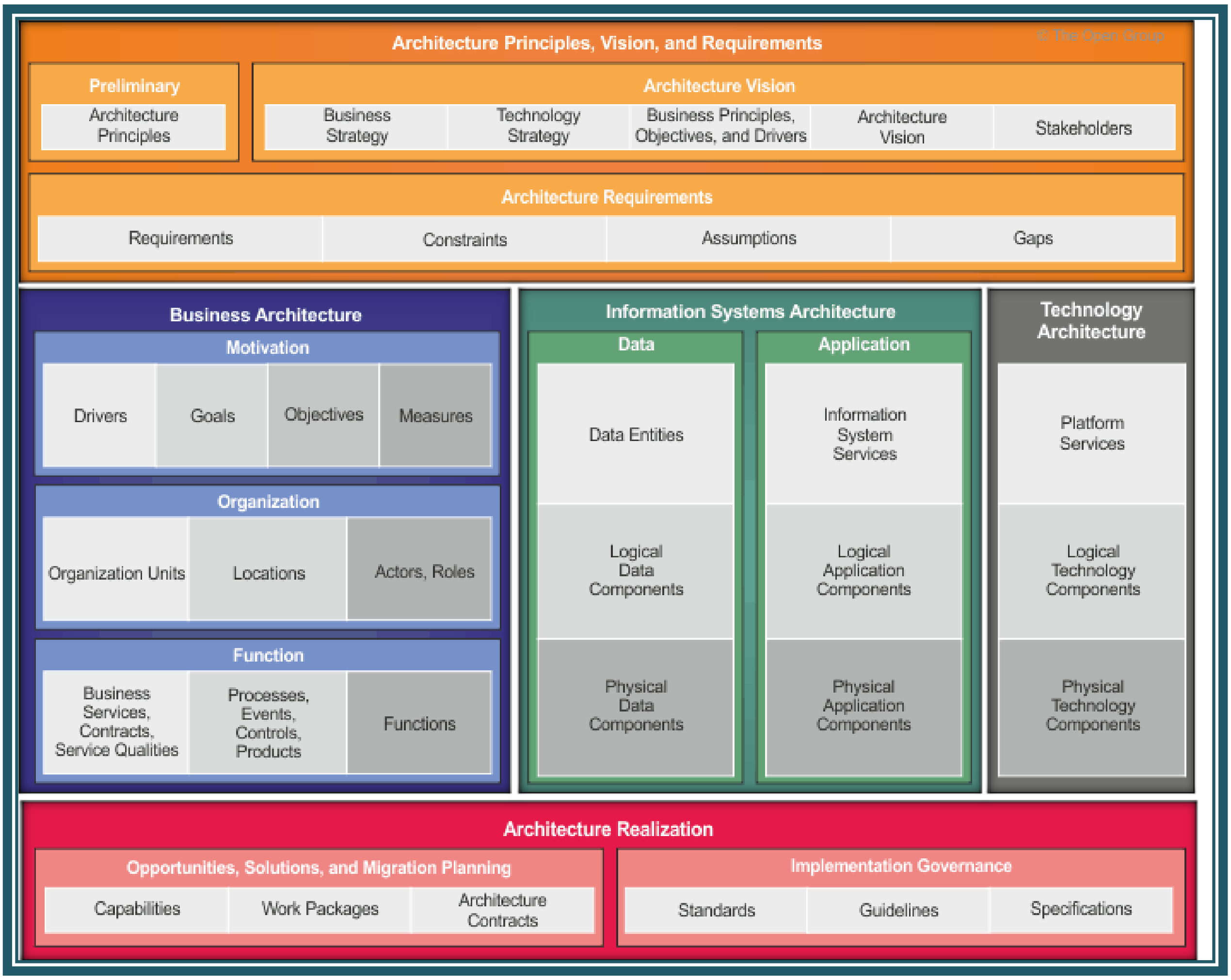


Figure 6-16: TOGAF Content Metamodel (The Open Group, 2009a)

- Enterprise Continuum and Tools – a view of the architecture repository that shows all the different architectures (for different stakeholders and different requirements), its relationships and its evolution. Figure 6-17 depicts the components of the enterprise continuum.

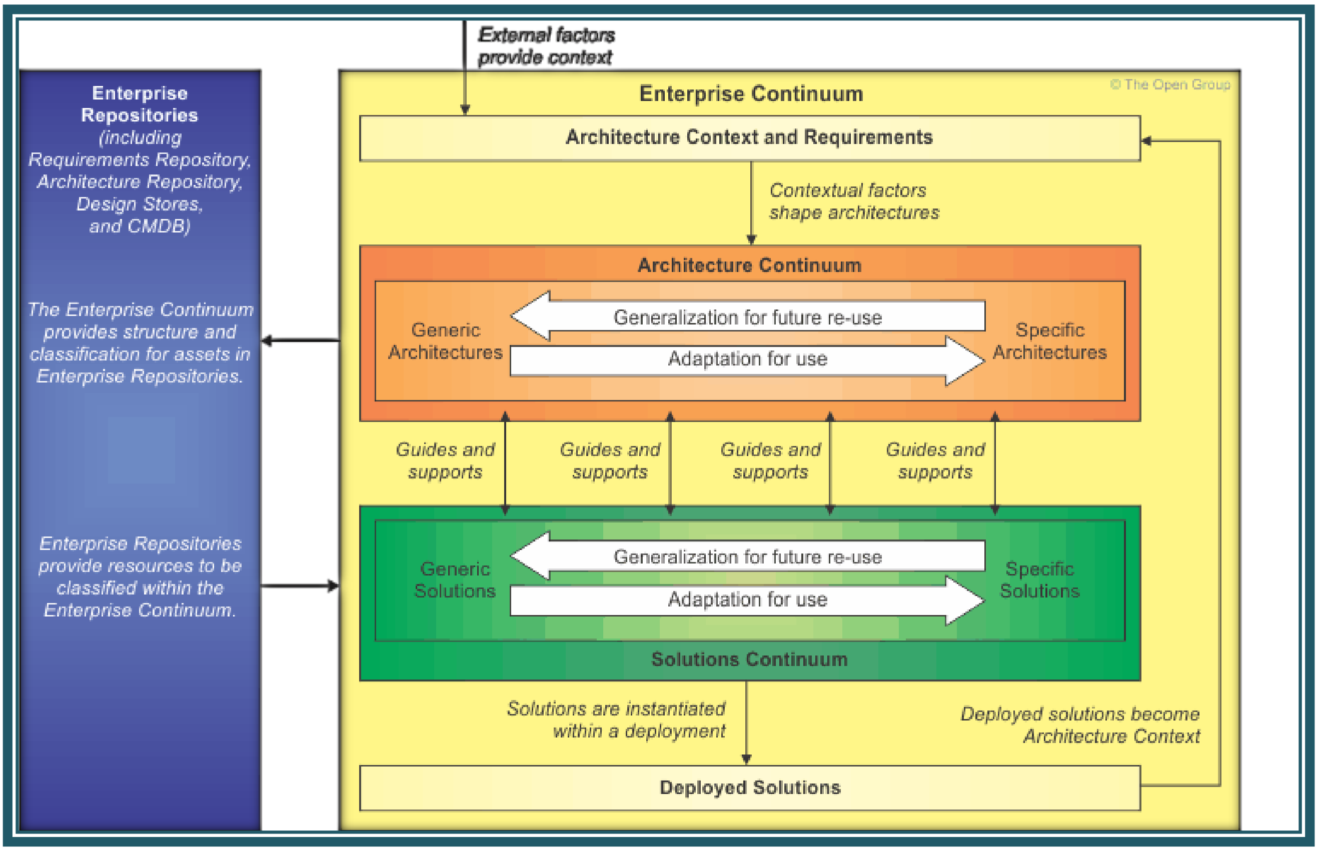


Figure 6-17: TOGAF Enterprise Continuum (The Open Group, 2009a)

- Reference Models (TRM) – a taxonomy (which defines terminology and a coherent description) and an associated TRM graphic (which provides a visual representation of the taxonomy). This includes the Integrated Information Infrastructure Reference Model (III-RM) – focusing on the application software space, the "Common Systems Architecture", business applications and infrastructure applications parts.
- Architecture Capability Framework – a set of reference materials to put in place appropriate organisation structures, processes, roles, responsibilities, and skills to realise an architecture capability successfully.

Some TOGAF documentation (The Open Group, 2009c: 63) names the integration strategy a meta-architecture framework. The purpose of this meta-architecture framework is to

- assist the architects to understand how the components fit into the framework;
- "derive the architectural models that focus on enterprise-level capabilities"; and to
- "define the conformance standards that enable the integration of components for maximum leverage and re-use".

Figure 6-18 illustrates how the different architectural domains need to co-exist. There are three dimensions: the vertical dimension (y-axis) indicates the enterprise depth, the horizontal dimension indicates (x-axis) the scope and the depth dimension (z-axis) the different architectural domains. “Key factors to consider are the granularity and level of detail in each artefact, and the maturity of standards for the interchange of architectural descriptions” (The Open Group, 2009c: 64). Projects or focuses within an enterprise, such as SOA, universal data models or integrated information infrastructure, will facilitate integration toward the higher end of the spectrum.

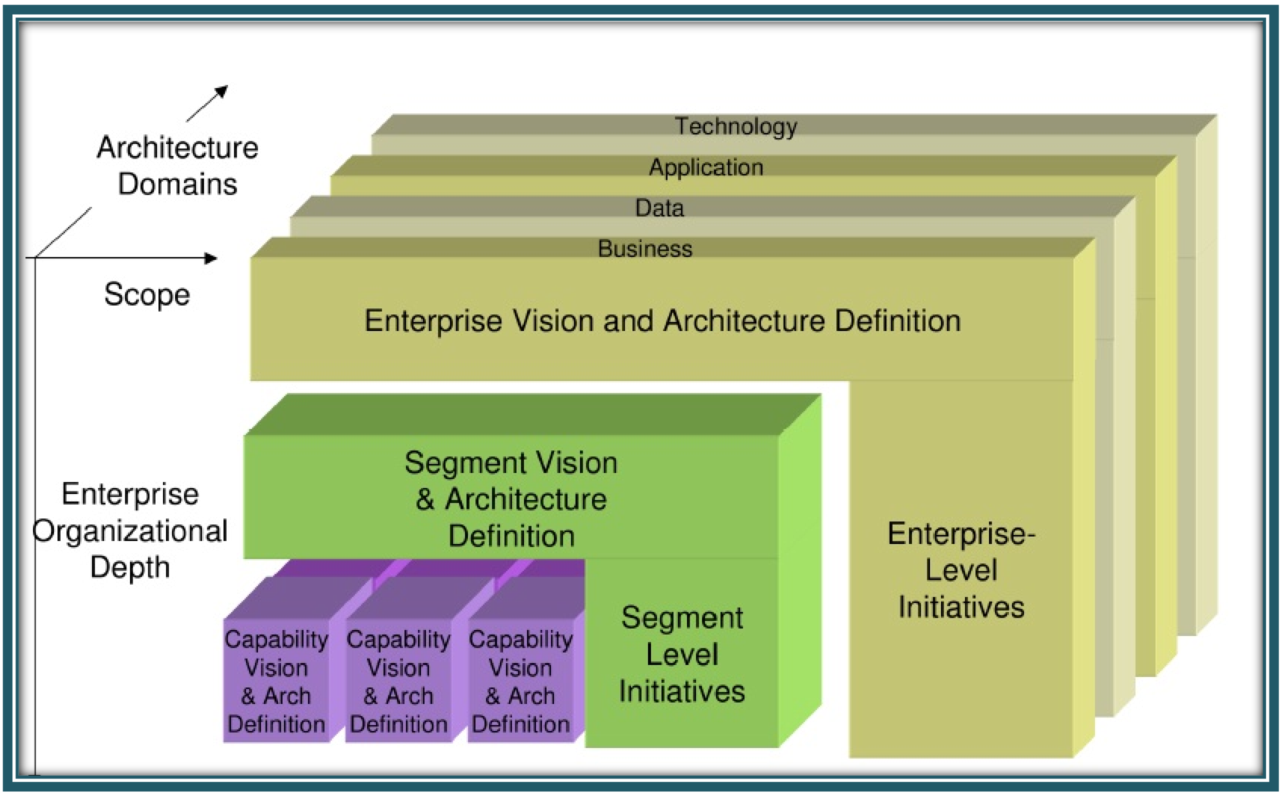


Figure 6-18: TOGAF's Integration of Architecture Artefacts (The Open Group, 2009c: 64)

Gerber, Kotzé & Van der Merwe (2010: 58) propose the use of ontologies to enhance the metamodel integration of TOGAF. They applied this through a process of identifying the concepts and concept hierarchy, identifying the disjointed steps, adding relationships, definitions, annotations and reiteration of these steps. An example of the concept hierarchy is depicted in Figure 6-19.

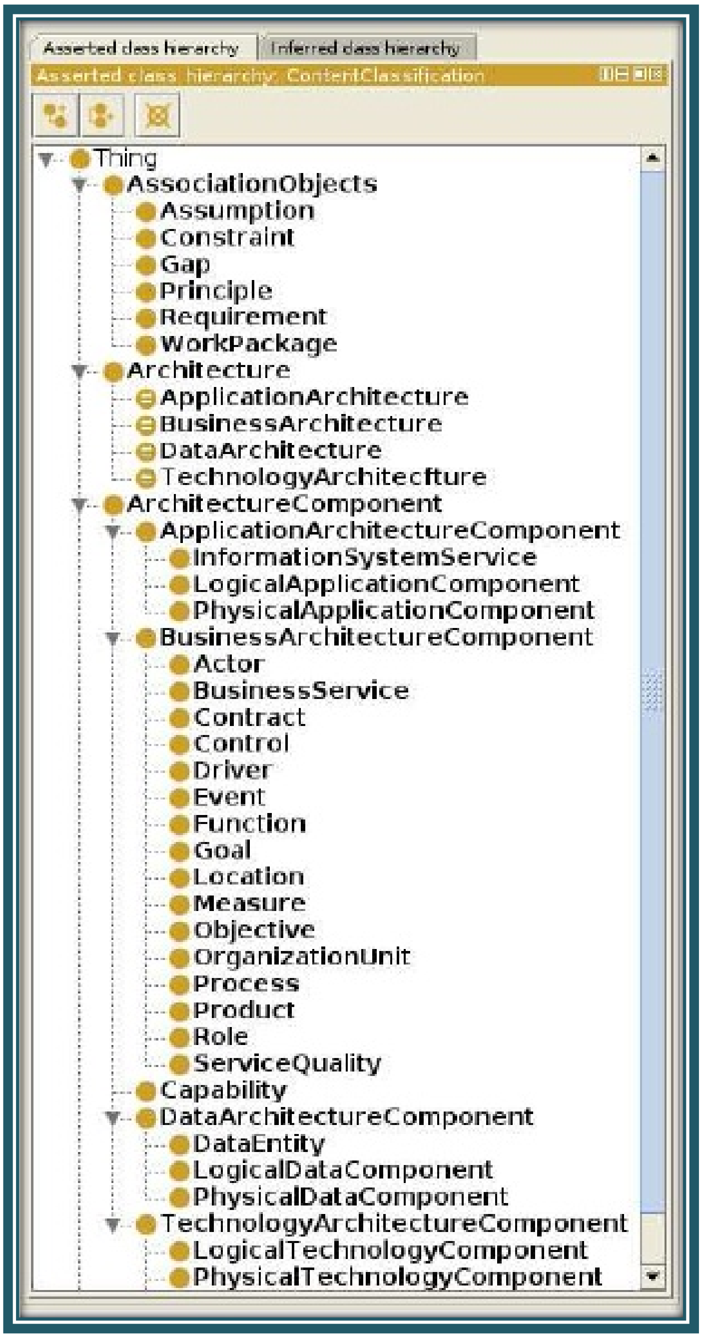


Figure 6-19: Example of Concept Hierarchy by Gerber, Kotzé & Van der Merwe (2010: 60)

TOGAF thus establishes relation between the different architectural domains through the use of the integration strategy of an overarching metamodel (meta-architecture), including a taxonomy, common language and reference architecture. The interlinking mechanisms can be enhanced by making use of ontologies (Gerber, Kotzé & Van der Merwe, 2010: 58).

6.5.1.3 <u>DoDAF</u>

DoDAF builds on three sets of views, namely: operational view, systems and services view and technical standards view (Urbaczewski & Mrdalj, 2006: 19). There is a fourth view, called the 'All View' (see Figure 6-20), which provides

- the linkage between the views by means of a dictionary/vocabulary that defines terms, taxonomy and metadata;
- context and scope, for example, subject area and time frame;
- the setting in which the architecture exists and includes the interrelated conditions, for example, tactics, doctrine, techniques, goals, vision and scenarios; and
- overarching aspects that relate to all three views and can be used as a planning guide (USA. Department of Defence, 2007a: 1-9; 2007b: 3-1).

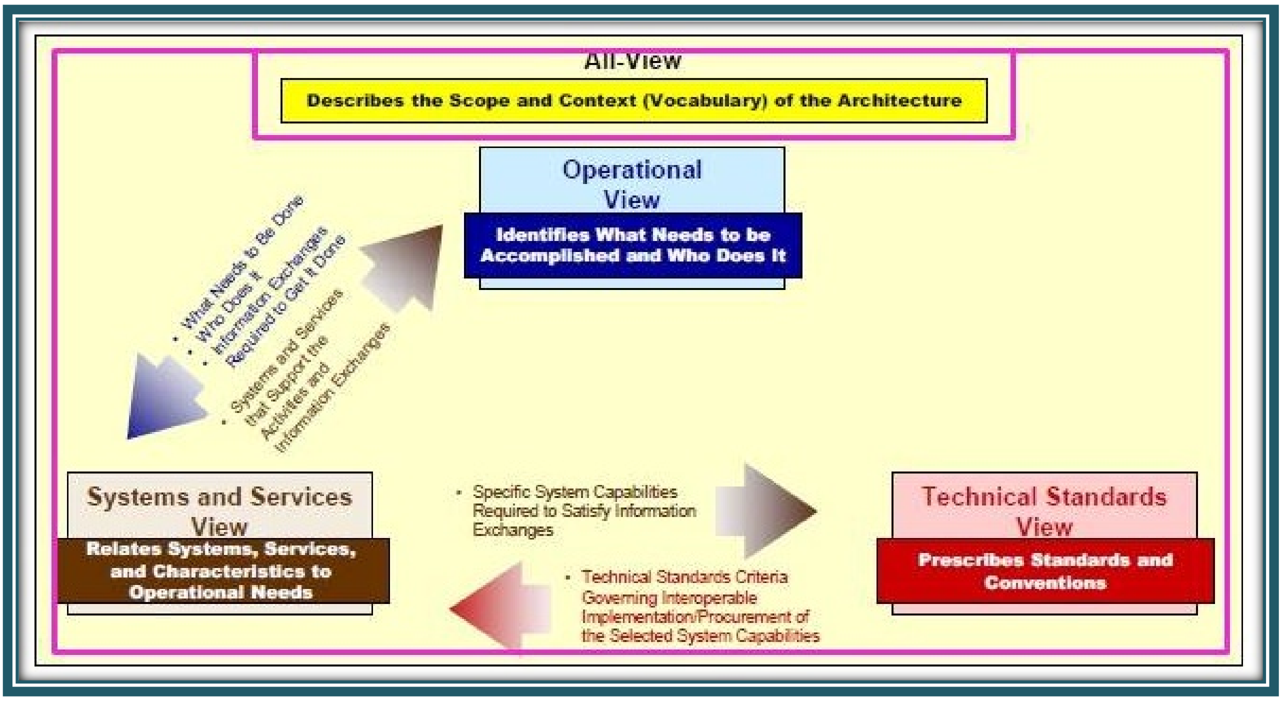


Figure 6-20: DoDAF v1.5 All view and Linkages (adapted from USA. Department of Defence, 2007a)

Some changes were brought about in DoDAF v2.0 as illustrated in Figure 6-21. The Capability Viewpoint, Project Viewpoint and Data and Information Viewpoint were added. The Technical Standards Viewpoint was renamed to Standards Viewpoint. The Systems and Services View was divided into two separate viewpoints.

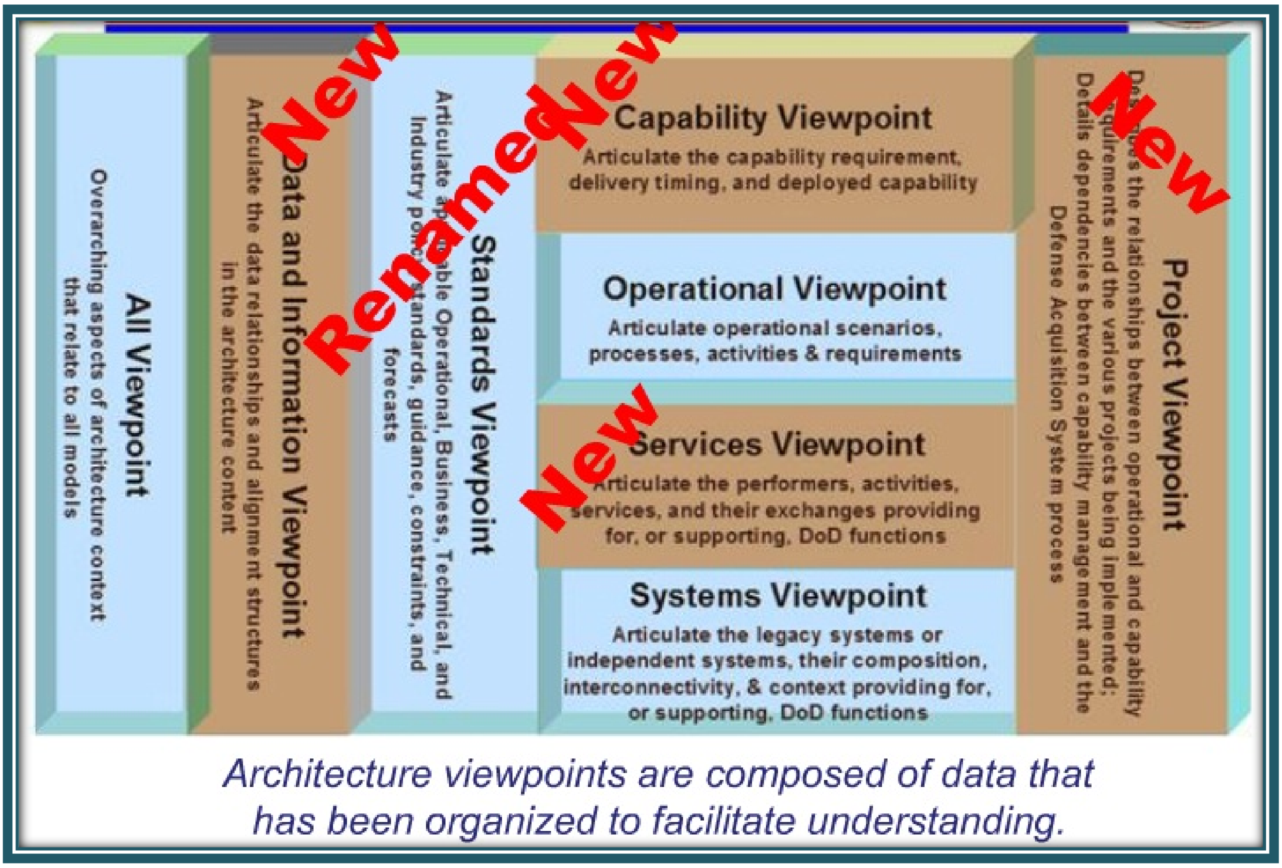


Figure 6-21: DoDAF v2.0 Architecture Viewpoints (Okon, 2010: 9)

The CADM (Core Architecture Data Model) is used to define a standard set of architecture data entities and relationships for architecture data (USA. Department of Defence, 2007b: 1-3). Each view of the architecture is specified as a separate instance of 'Architecture' and the collection of views are related to the overall instance of 'Architecture'. Thario (2008) describes the format for data exchange between repositories as a unique strength of DoDAF. This data exchange format facilitates data exchange between repositories and tools in order to manipulate the architectural artefacts. With each view it defines the data interchange requirements and format to be used when exporting the data into the common format.

DoDAF aims to make all their products and operations net-centric, that is: "enabling the ability to share information when it is needed, where it is needed, and with those who need it" (USA. Department of Defence, 2007b: 2-12). The DoD uses it to translate information superiority into combat power by the effective linking of well-informed entities. DoDAF assists the process by supplying net-centric concepts and constructs within all the views and domains.

DoDAF's integration strategy is thus to make use of a metamodel enabled by viewpoints, metadata and taxonomies to address the integration of the architectural domains. A net-centric common language forms part of the interlinking mechanism.

6.5.1.4 FEA

FEA was developed by the USA's federal government as a comprehensive standard framework for all government bodies. "The FEAF provides a structured approach to federal organizations to integrate their respective architectures into the federal enterprise architecture (FEA)" (Saha, 2007: 3). This allows for the sharing of data, code and processes. FEA (see Figure 6-22) consists of (Sessions, 2008: 7)

- a segment (cross-agency business areas) model;
- a set of reference models (business, service, components, technical and data) incorporated into a Consolidated Reference Model (USA. Federal Chief Information Officers Council, 2010: 1);
- an architectural process;
- a taxonomy for describing assets; and
- a maturity model.

It makes provision for business architecture, data architecture, technology architecture and application architecture (Saha, 2007). FEA also provides three general profiles "which are intended to promote common, consistent enterprise architecture practices that improve government performance" (USA. Federal Chief Information Officers Council, 2010). The FEA profiles (enterprise services in Figure 6-22) include a

- Geospatial Profile,
- Records Management Profile, and
- Security and Privacy Profile (FEA-SPP).

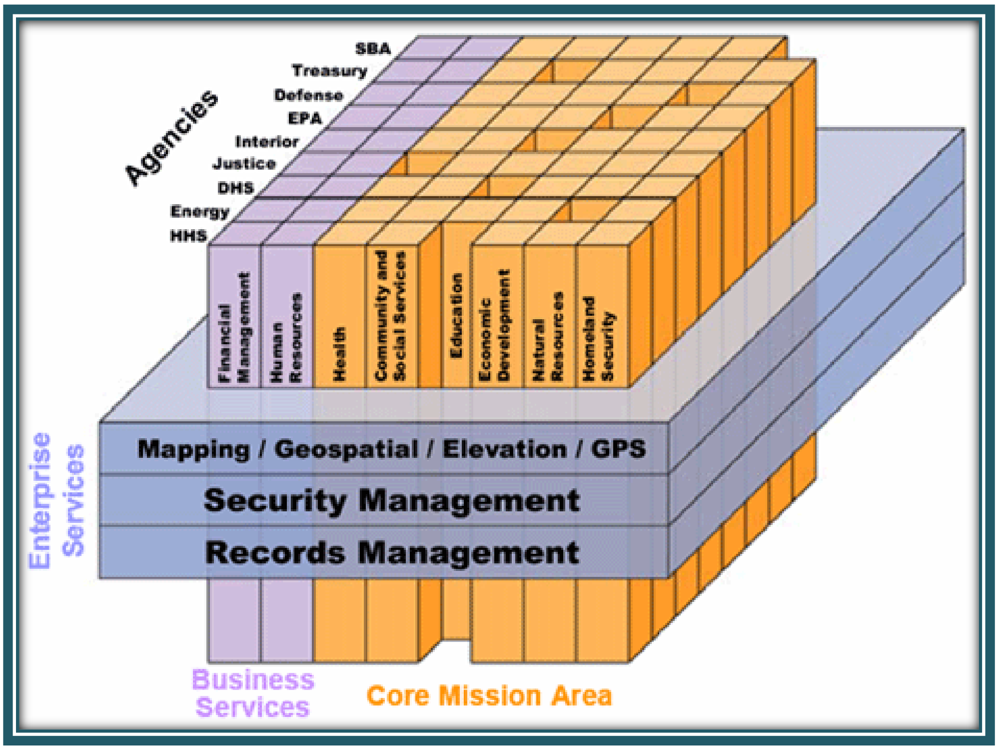


Figure 6-22: FEA Segment Model (Sessions, 2007)

The structure and methodology of FEA are depicted in Figure 6-23. The major components are:

- architecture drivers – represent external stimuli that cause changes;
- strategic direction – ensures changes are consistent with enterprise strategy;
- current architecture – indicates current state;
- target architecture – indicates the target state;
- transitional processes – applies changes from the current architecture to the target architecture in compliance with the standards;
- architectural segments – focus on a subset or smaller enterprise within the bigger enterprise;
- architectural models – provides the documentation as well as the basis for managing changes; and
- standards – includes standards, guidelines and best practices (Schekkerman, 2004b: 108).

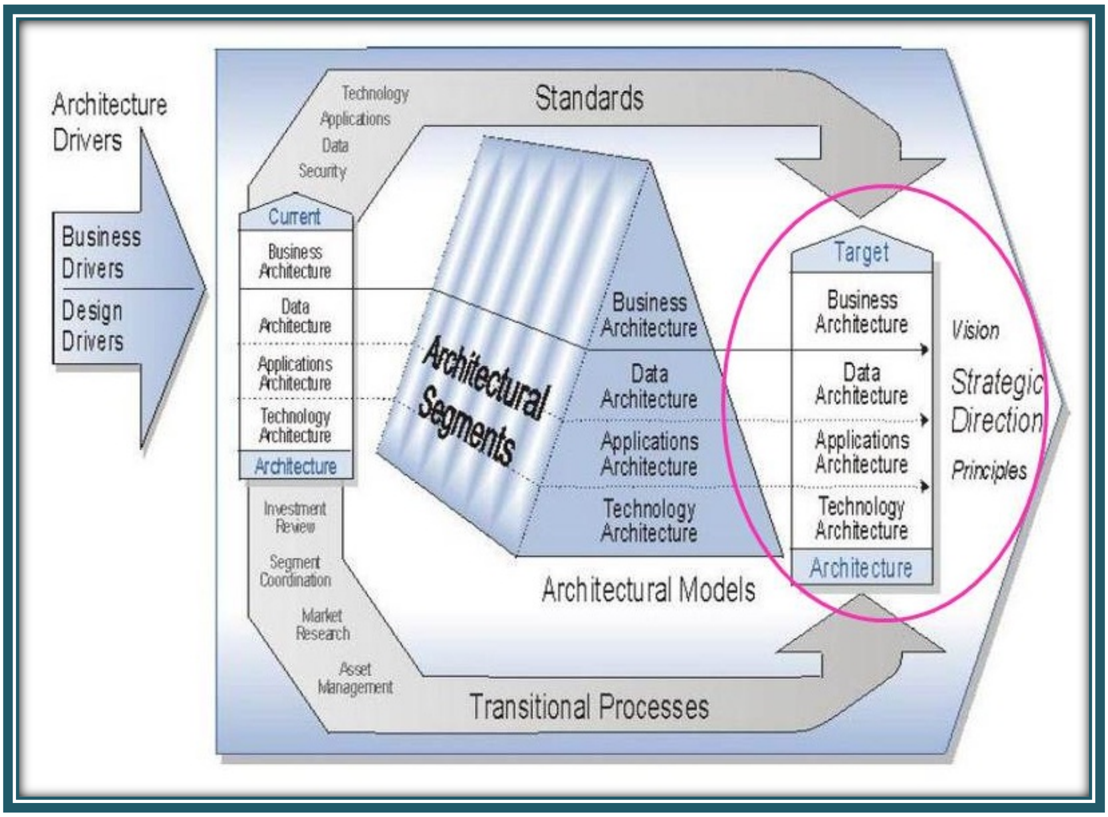


Figure 6-23: The Structure and Methodology of FEA (adapted from USA.Chief Information Officer Council, 2001: 26)

FEA is being used by a variety of agencies and business units, divided into multiple segments. Enterprise services are implemented across multiple segments. The re-use of shared assets is classified at an enterprise level and aligns the segment architectures with the business as well as external areas of collaboration (De Vries, 2010: 25).

FEA makes use of a metamodel strategy for integration and re-use of shared assets like general profiles, standards, processes and the overarching Consolidated Reference Architecture. It contains transitional processes, such as investment review, segment coordination, market research and asset management, for interlinking mechanisms.

6.5.1.5 <u>Gartner Enterprise Architecture Method (GEAM)</u>

There are two main parts of the GEAM, namely the Gartner Enterprise Architecture Process Model and the Gartner Enterprise Architecture Framework (GEAF) (James *et al.*, 2005: 2). The GEAF makes use of architectural viewpoints and an enterprise context layer. The context layer contains the articulation of the enterprise's business strategy and its implications as well as the external influences. The primary viewpoints addressed are: enterprise business architecture, enterprise information architecture and enterprise technology architecture. Additional viewpoints may be extracted when there is a specific stakeholder requirement for it, for example, a compliance viewpoint (James *et al.*, 2005: 3).

Gartner also makes use of the Enterprise Solution Architecture (ESA) as a synthesis of the other architectural domains and to articulate the relationships among them (De Vries, 2010: 25). "An ESA is a consistent architectural description of a specific enterprise solution. An ESA combines and reconciles the requirements, principles and models of intersecting stakeholder-specific viewpoints into a complete architectural description of a specific enterprise solution" (James *et al.*, 2005: 5).

This is a unique concept to address "the single most important and challenging architectural issue: combining and reconciling the loosely coupled and often conflicting viewpoints of the primary stakeholders into a unified architecture for an enterprise solution" (James *et al.*, 2005: 3). The ESAF is an architectural description of how to create an ESA. The ESAF is designed to reunite the divided viewpoints and is essentially a meta-architecture to generate the portfolio of enterprise solutions. The ESAF focuses the attention on the dependencies and inconsistencies among the other architectural viewpoints. ESA has three major areas of focus: individual solution architecture, repeatable solution pattern models and reusable solution portfolios (Guevara & Robertson, 2011: 2). Figure 6-24 depicts the different components of ESA in the GEAF context.

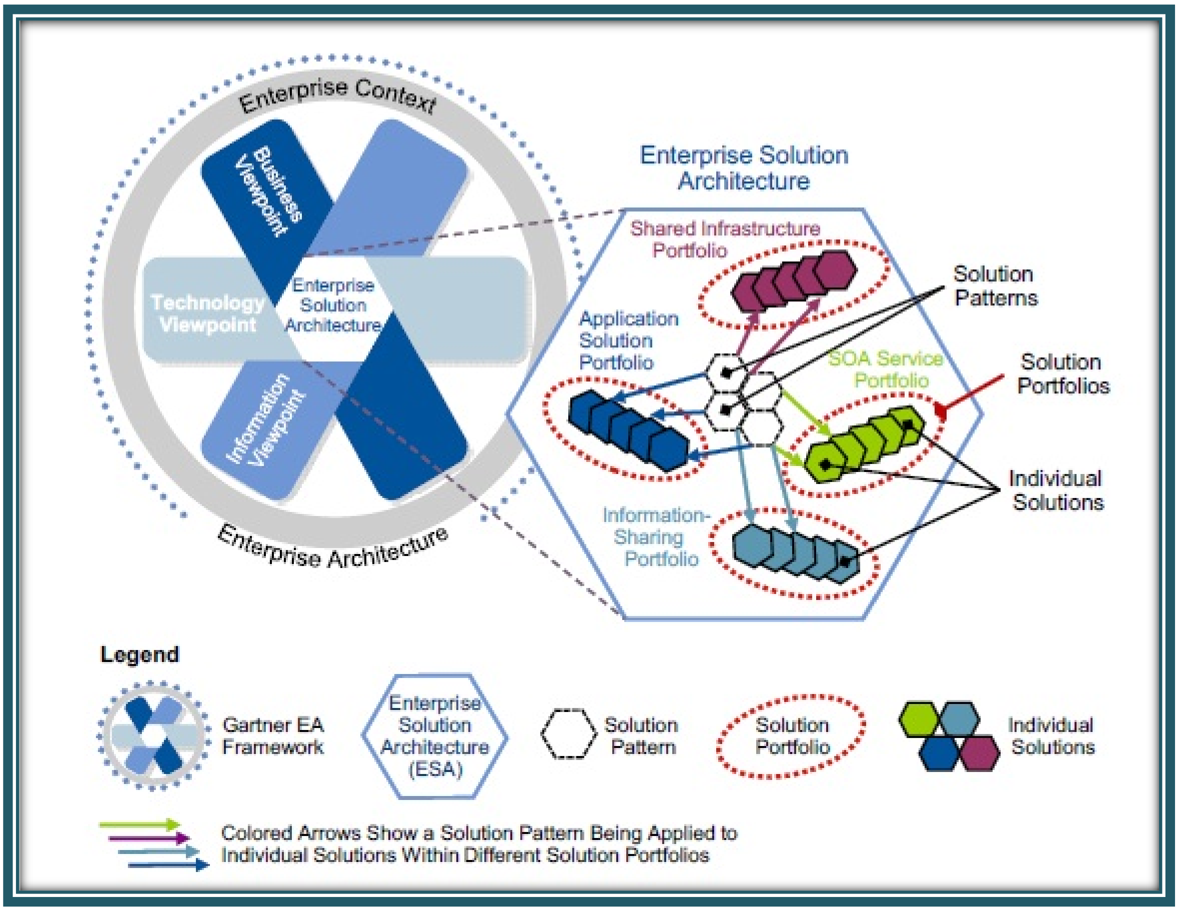


Figure 6-24: Gartner Enterprise Solution Architecture (Guevara & Robertson, 2011: 8)

The integration strategy of the GEAF thus addresses different viewpoints which are reunited/reconciled (integrated) through the Enterprise Solution Architecture, which also creates reusable entities. The Gartner Enterprise Architecture Process Model provides methodology and processes as the interlinking mechanisms.

6.5.2 Other Frameworks Illustrating Relationships

Lankhorst (2004: 210), based on work by Creasy & Ellis (1993), states that the problem of interlinking between different architectures/models can be solved through two different approaches:

- Define a direct mapping between each pair of modelling languages in order to obtain direct relations.
- Use a core conceptual language as an intermediary language, which would require fewer mappings than the first approach.

The following frameworks have specific integration strategies or illustrate interlinking mechanisms between architectural domains.

6.5.2.1 Domain Architecture

Bruls *et al.* (2010: 518) introduce a concept, which they call "domain architectures". This is an additional type of artefact limited to a certain scope. They propose a separate architect(s) to be responsible for refining the architecture. While the enterprise architecture, business, information, IT and solution architectures address the scope for a particular part of the enterprise, "the scope of a domain architecture addresses a level in-between." The term 'domain' is used here with a different meaning than in the rest of the literature.

Partitioning is necessary to define the concept of domain architecture. Figure 6-25 depicts this partitioning at different levels of detail and with different scopes. "The figure positions partial enterprise architecture, domain architecture and solution architectures on different places on a refinement axis (running transversal through the picture from lower left to upper right" (Bruls *et al.*, 2010: 519). Example scopes are illustrated in the figure. The scope decreases as the level of detail increases on the refinement axis.

"Domain architectures are differentiated from enterprise architectures by the fact that they include an element of detailing and reduced scope that are the result of a more fine-grained viewpoint and an increased focus on engineering of solutions" (Bruls *et al.*, 2010: 520). Partitioning is also used by *inter alia* Ross, Weill & Robertson *(2006)* and Versteeg & Bouwman (2006) to differentiate between different layers of enterprise architecture, like business architecture and IT architecture. Although these layers could also be called domains (Iyer & Gottlieb, 2004), Bruls *et al. (2010: 520)* view them as 'partial enterprise architectures', used to make enterprise architecture more manageable, but operating on the same level of detail as enterprise architecture.

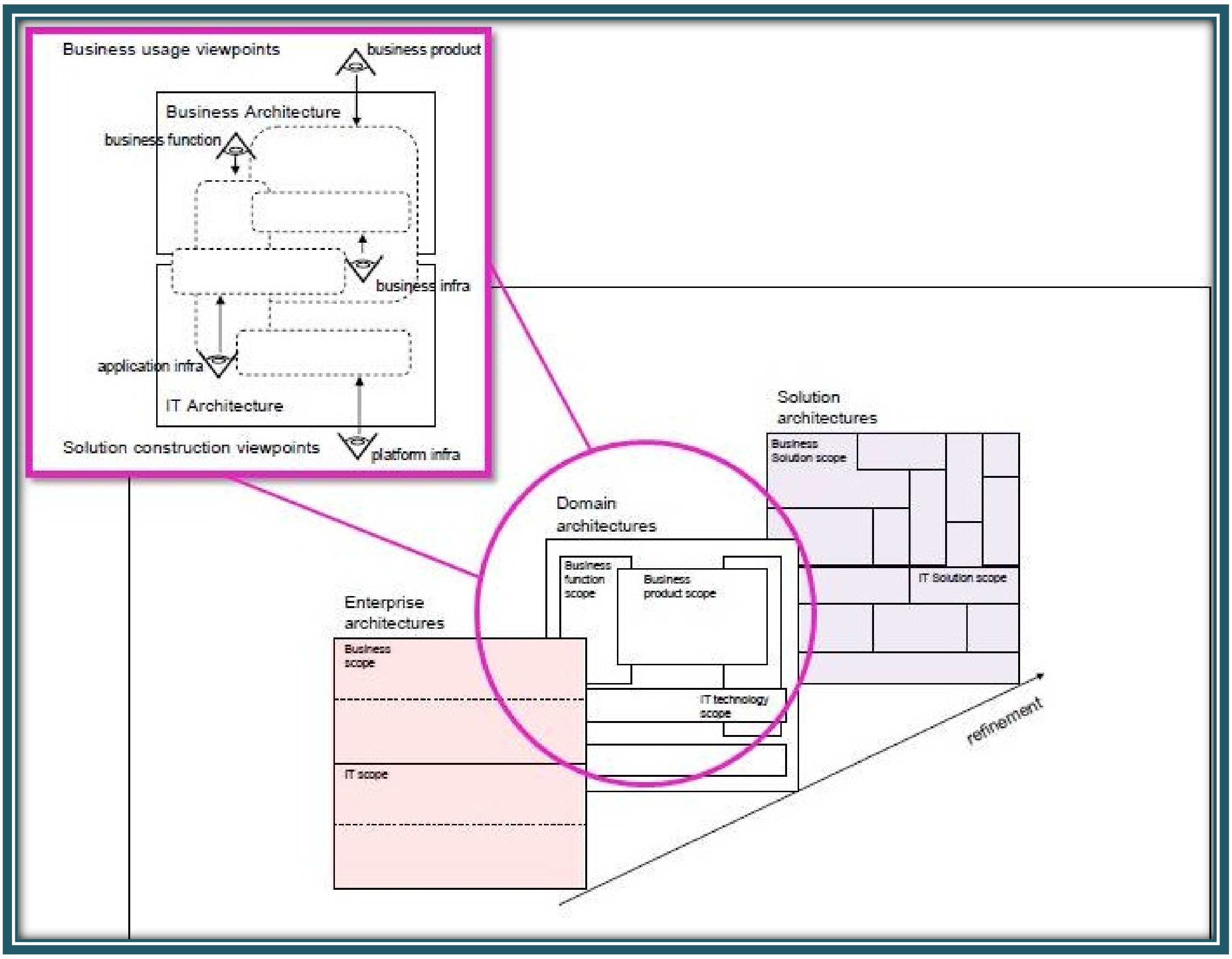


Figure 6-25: Domain Architecture Partitioning (adapted from: Bruls *et al.*, 2010: 519)

The emphasised insert in Figure 6-25 shows the combination of the business usage viewpoints (looking from the top of the figure) and the solution construction viewpoints (looking upward from the bottom of the figure). 'Usage' and 'construction' are the keywords for the viewpoints that need to be aligned and linked in both directions. "The usage and construction perspectives are layered perspectives that can consider the solution space from several levels as indicated visually in the figure" (Bruls *et al.*, 2010: 524). Based on this principle of viewpoint categorisation, a domain taxonomy was constructed, as depicted in Figure 6-26.

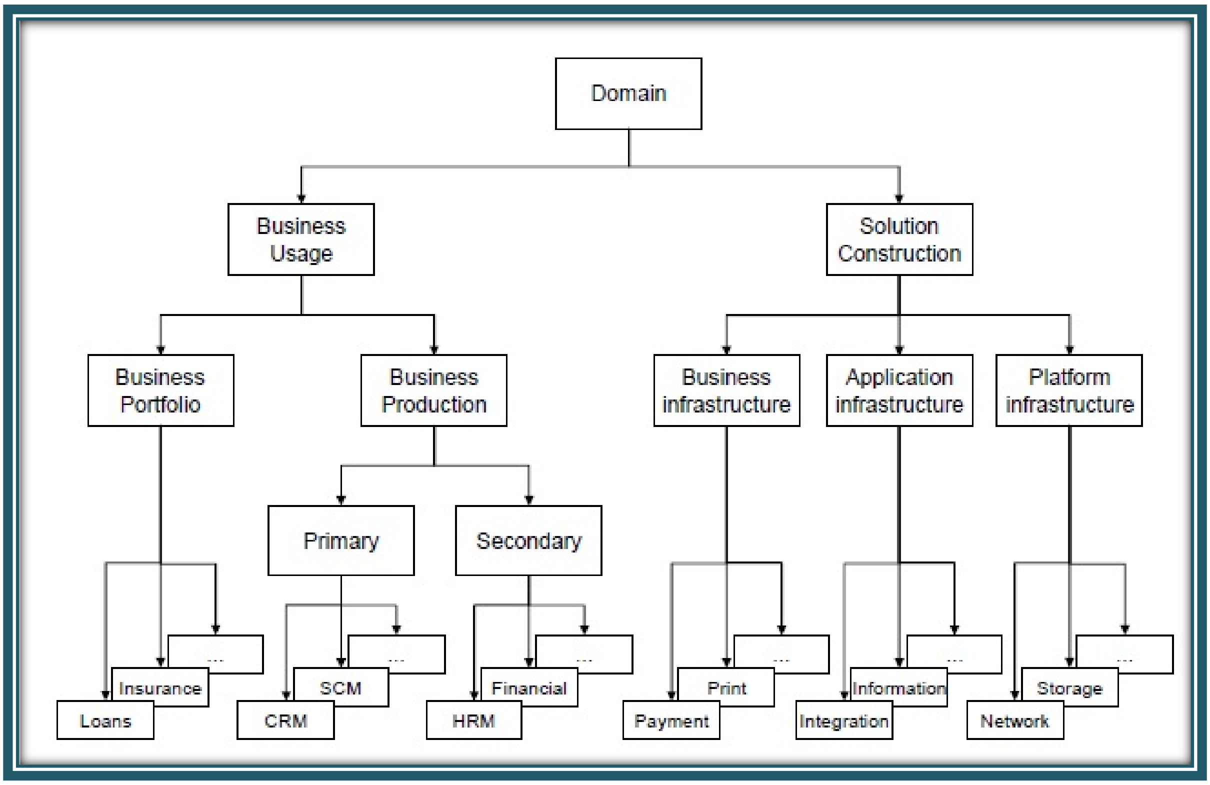


Figure 6-26: Domain Taxonomy (Bruls *et al.*, 2010: 525)

Figure 6-26 depicts three levels of domain categorisation and the leaf level of domain architecture instances. The main branch is between the business usage and the solution construction. Business usage is then divided into portfolio and production, while solution construction is divided into business, application and platform infrastructure. The leaves provide examples for a financial enterprise.

The guidelines for the implementation of domain architecture are:

- relevance governance – domains should be focused on the needs of a stakeholder to ensure its importance and relevant contents;
- ownership governance – domains should be controlled by an authority to ensure clear responsibilities for the execution obligations;
- bounded content – domains should have clear boundaries to allow clear scope of responsibility;
- cohesive content – domains should have related contents with a focus that differentiates from other domains to allow work in a domain to proceed independently and provide well-structured output;
- composable content – “domains should provide results that are composable with that

of others towards engineering" to allow the combination of results into an overall engineering guidance (Bruls *et al.*, 2010: 522).

Classification guidelines are tabled in Table 6-2.

Aspect	Business usage domain architecture	Solution construction domain architecture
Scope	Support a selected *part of the full* business and IT operating environment	Support *(a large part of) the full* business and IT operating model
Viewpoint	Created from the *usage viewpoint* (business product or function)	Created from the *constructional viewpoint* (business infra, application infra, platform infra)
Functionality	Business functionality directly used in business operations	Infrastructure functionality indirectly used in business operations
Consisting of	Architecture building blocks that *work together* to provide a solution	Architecture building blocks that *play a similar role* across solutions
Selection	*proximity* based selection	*similarity* based selection
Stakeholder	Enterprise roles with *responsibility for a specific part* of the business	Enterprise roles with a *responsibility for construction infrastructure*

Table 6-2: Domain Classification Guidelines (Bruls *et al.*, 2010: 528)

The integration strategy of domain architecture is to make use of finely grained viewpoints called domains. These domains are interlinked through partitioning by aligning the usage and construction aspects of the relevant domains.

6.5.2.2 Service Orientation

Service orientation leads "to a layered view of enterprise architecture models, where the service concept is one of the main linking pins between the different layers" (Lankhorst *et al.*, 2005: 86). The layered view is depicted in Figure 6-27. It contains service layers interleaved with implementation layers and may also include internal services. The layers are linked through 'used by' relations, for example, the applications make use of the infrastructural layer below. It also uses 'realisation' relations, for example, how services are realised in an implementation layer.

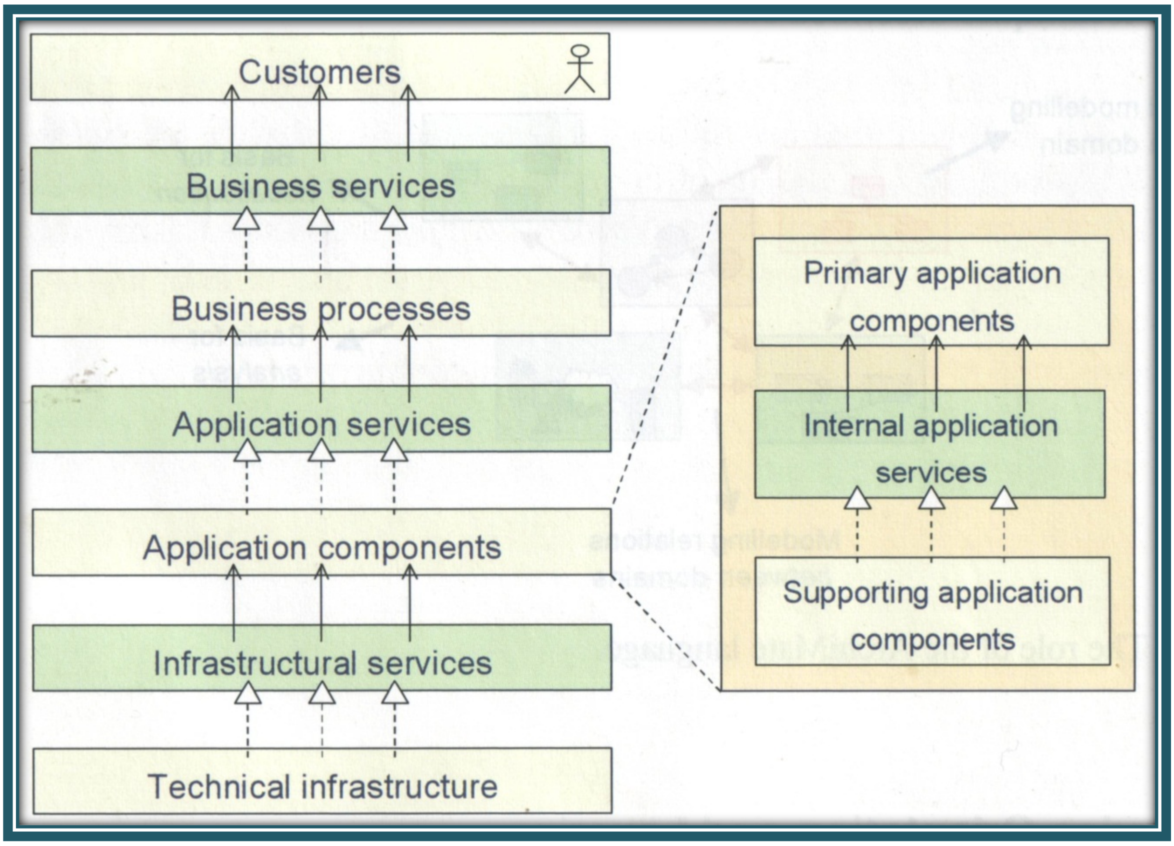


Figure 6-27: Service Layered View (Lankhorst *et al.*, 2005: 86)

Service Orientated Architecture (SOA) is an example and aims to be business-driven, by

- creating a portfolio of capabilities representing business, information and technology concepts;
- composing or orchestration of these capabilities (services) "along with events, rules, and policies into business processes and solutions that fulfil business scenarios";
- focussing on and working towards a business outcome – in other words execution for business reasons (SOA Consortium EA2010. Working Group, 2010: 2).

The SOA Consortium illustrates a possible integration in Figure 6-28. It builds a linkage between the business processes component of business architecture and the use cases component of technology architecture. They do, however, also propose the following linkages:

- business strategy and IT strategy;
- business architecture and technology architecture; and
- business requirements and business solution delivery (IT).

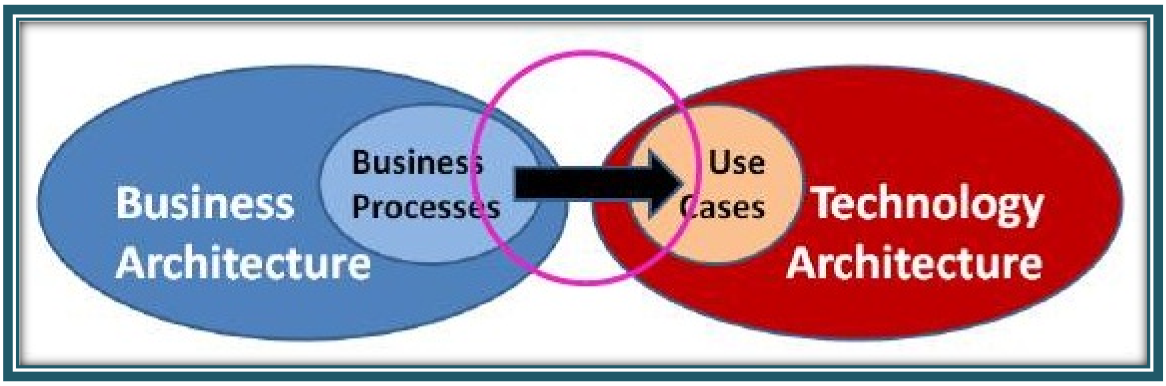


Figure 6-28: BA and TA linking (adapted from:SOA Consortium EA2010. Working Group, 2010: 4)

Bygstad & Aanby (2010: 258) regard the enterprise service bus (ESB) as an implementation of SOA, that allows for full integration between internal and external services at a transactional level, but is also flexible and easy to adjust. They apply the ESB as an organisation structure, an innovation infrastructure and a technical infrastructure.

The integration strategy of service architecture is thus to make use of service layers. These are interlinked through the process of building relations between the service layers. Example relations are ‘used by’, realisation, orchestration and capabilities.

6.5.2.3 Viewpoints

A possible method to integrate the diverse architecture descriptions is to follow an approach where architects/stakeholders define their own views of the enterprise architecture – these views are specified by ‘viewpoints’. “*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” (Lankhorst *et al.*, 2005: 147). Viewpoints are used to view aspects in isolation as well as to view the relationship between two or more aspects. The concept of viewpoints forms part of the IEEE 1471 standard for architecture description (IEEE, 2000: 4). Aier *et al.* (2009: 38) describe viewpoints as “situational fragments of the EA model”. A number of the enterprise frameworks utilise viewpoints, like the Zachman Framework, RM-ODP and TOGAF.

Lankhorst *et al.* (2005: 152) describe a viewpoint approach, where the content and presentation is separated from the visualisation of a view. Different visualisation techniques can thus be used on the same modelling concepts and *vice versa*. Furthermore operations on the visualisation of a view, like layout changes, need not change its content. "The creation and update of both the view and the visualisation are governed by a viewpoint", which is jointly defined/selected by the architects and stakeholders through an iterative process. This is depicted in Figure 6-29 and illustrates the separation of the different concerns: model, view, visualisation and viewpoint.

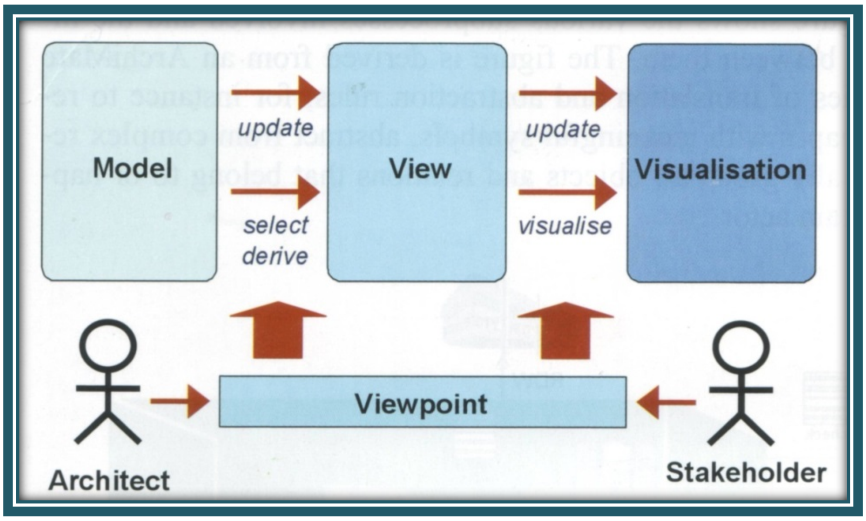


Figure 6-29: Separation of Model, View, Visualisation and Viewpoints (Lankhorst *et al.*, 2005: 153)

Lankhorst *et al.* (2005: 161) also provide a classification of different architecture viewpoints, as depicted in Figure 6-30.

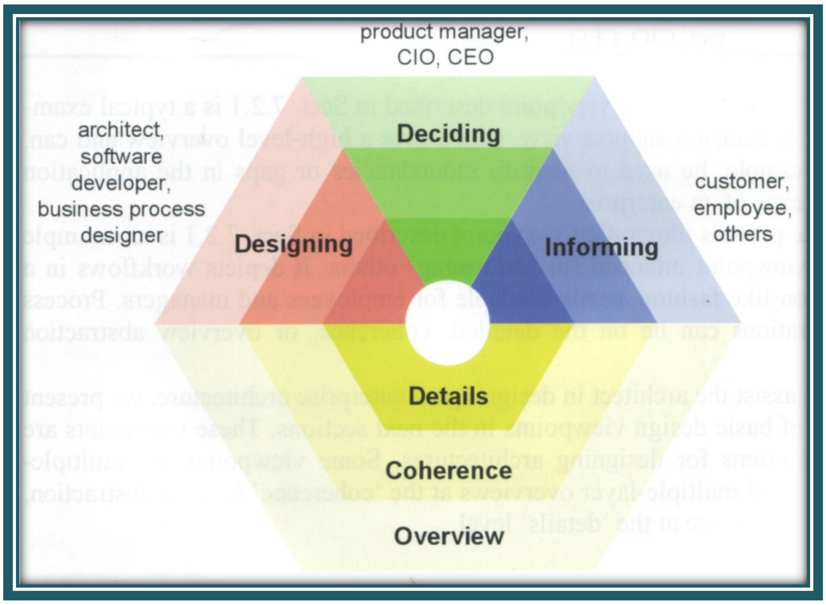


Figure 6-30: Classification of EA Viewpoints (Lankhorst *et al.*, 2005: 163)

The top half of the diagram focuses on the *purpose* dimension, providing the three categories as well as possible role players on different levels:

- designing – to support the different levels of designers in the whole design process (initial sketch to detailed design), for example, diagrams;
- deciding – to support the different levels of managers in decision-making, for example, cross-reference tables, landscape maps and reports;
- informing – to support the different stakeholders in communication in order to achieve understanding, obtain commitment, and convince adversaries, for example, illustrations, animations and flyers (Steen *et al.*, 2004).

The bottom half of the diagram focuses on the *content* on three levels of abstraction, namely:

- details – spanning one layer and one aspect of the framework, for example, a BPMN process diagram or a UML class diagram;
- coherence – spanning multiple layers or multiple aspects of the framework, for example, process-use-system or application-uses-object;
- overview – spanning both multiple layers and multiple aspects of the framework, for example, overviews for decision-makers like CEOs (Chief Executive Officer) and CIOs (Lankhorst *et al.*, 2005: 163).

The ArchiMate language, for example, proposes or makes use of a whole range of possible viewpoint types:

- Introductory Viewpoint
- Organisation Viewpoint
- Actor Cooperation Viewpoint
- Business Function Viewpoint
- Product Viewpoint
- Service Realisation Viewpoint
- Business Process Cooperation Viewpoint
- Business Process Viewpoint
- Information Structure Viewpoint
- Application Cooperation Viewpoint

- Application Usage Viewpoint
- Application Behaviour Viewpoint
- Application Structure Viewpoint
- Infrastructure Viewpoint
- Infrastructure Usage Viewpoint
- Implementation & Deployment Viewpoint (Lankhorst *et al.*, 2005: 173).

The integration strategy is thus to make use of particular viewpoints to determine the relevant EA components as well as to provide mechanisms to create integrations relevant to the particular target group of the viewpoint.

6.5.2.4 Metamodels

The use of metamodels is a mechanism to incorporate the different components of a model into a more integrated model through a common language.

The term 'meta' is a prefix meaning 'related to' or 'information about' (Business Dictionary, 2010). A direct translation of metamodel is thus information about the models. "Metamodels are abstractions that are used to specify characteristics of models" (Gerber, Kotzé & Van der Merwe, 2010: 54). Metamodels are a part of and play an important role in EA by providing a common language for the enterprise. Some frameworks propose their own metamodels, for example, DoDAF and TOGAF, while others, like the Zachman Framework, are less metamodel-centric.

Saat *et al.* (2010: 18) propose the use of metamodels to accomplish a common language and an integrated view, by providing a template for the models. This metamodel prescribes the permissible entities, their attributes and their connections in order to achieve coherence between all models. It thus enforces semantic rigour among the models, which is a precondition for successful communication and documentation. The entity relations/connections can be made to correspond to a breakdown into causal relations used for predictions. Causal analysis is, however, problematic due to the great number of influential aspects and the complexity of the intertwined aspects. They suggest that an enterprise starts to focus on a few relevant scenarios and build up a set of metamodels as the situations change or require.

Saat *et al.* (2010: 16) populate a metamodel by first, delineating the design parameters and then second, enumerating the design parameters by using sets of qualities. These qualities are depicted in Figure 6-31. They make use of a jigsaw metaphor to indicate equal weight between the areas and to link the different pieces into one whole.

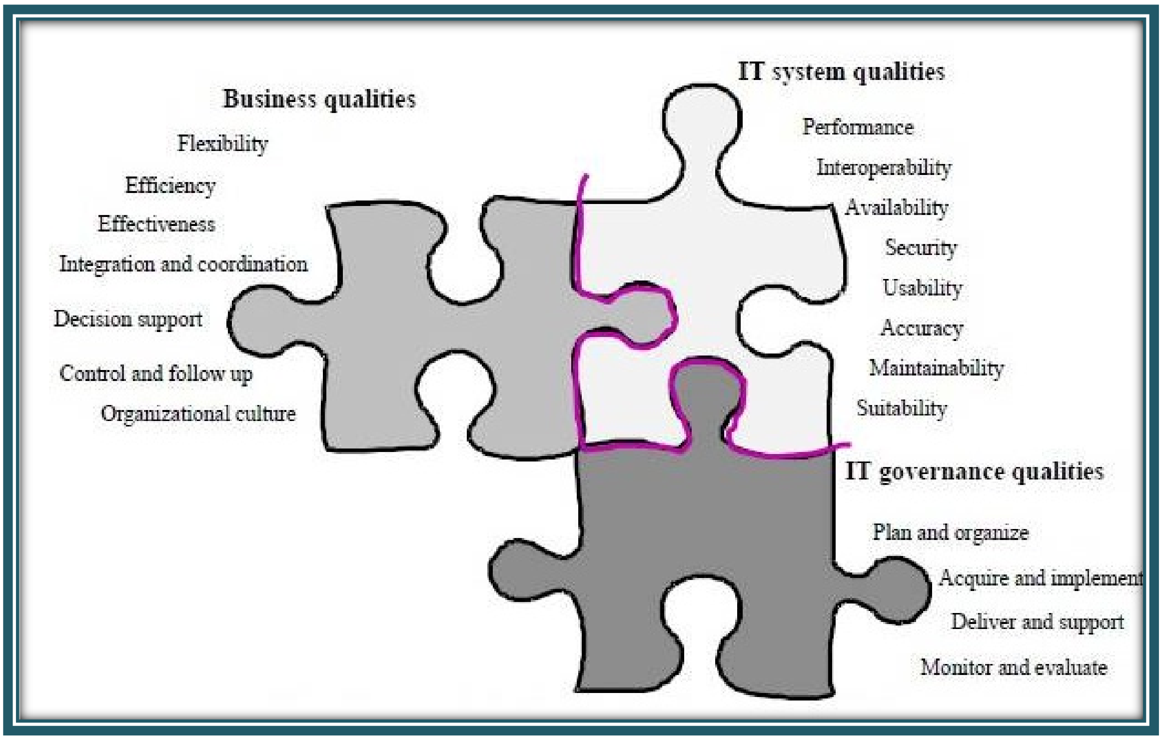


Figure 6-31: Conceptual View of an IT/Business Alignment Operationalization (adapted fromSaat *et al.*, 2010: 17)

Gerber, Kotzé & Van der Merwe (2010: 54) researched the metamodel for the Open Group's TOGAF 9, and found ambiguities and inconsistencies. They propose the use of ontology technologies to enhance the quality and consistency of the metamodel and conclude that ontologies "can represent the required information of metamodels but in a much more precise and unambiguous manner than that of metamodel notations currently used" (Gerber, Kotzé & Van der Merwe, 2010: 63).

Another example of a metamodel is the OMG's (Object Management Group) Meta-Object Facility (MOF). The MOF specification "is the foundation of OMG's industry-standard environment where models can be exported from one application, imported into another, transported across a network, stored in a repository and then retrieved, rendered into different formats (including XMI, OMG's XML-based standard format for model transmission and storage), transformed, and used to generate application code" (Anon., 2012b). MOF also contains mapping schemas. The principle is to create all the different models, frameworks and components of an enterprise's EA to be MOF-compliant. This ensures integration through the whole metamodel and architecture process.

The integration strategy is thus to make use of metamodels, which are interlinked through the process of using a common language, templates and reusable components.

6.5.2.5 EA Framework (Rohloff)

Rohloff (2008: 562) proposes an architecture framework (see Figure 6-32), based on the principal elements of the architecture for information systems described by Sinz in 1997. The framework contains

- domains – business architecture, application architecture and infrastructure architecture;
- views – each domain can be described according to the different views, *i.e.* component view (architecture elements and their relationships), communication view (interaction of elements) and distribution view (distribution of elements in terms of location or organisational assignment);
- dependencies – the framework proposes using blueprints to describe the relationship or dependencies between the architectural domains;
- standards – form an essential part of architectural building blocks to provide inter-changeability and ease of across-system communication; and
- pattern – identification and usage of commonly recognised pattern forms part of the objective for architecture design.

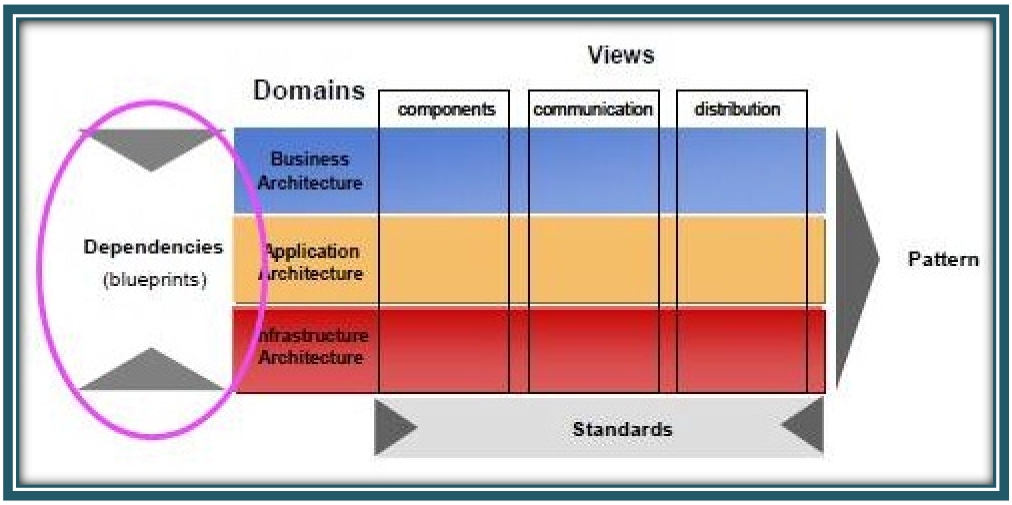


Figure 6-32: Enterprise Architecture Framework (adapted from Rohloff, 2008: 562)

Rohloff (2008: 564) advocates the use of blueprints to provide a comprehensive view of the different architecture building blocks and their interactions. Blueprints "show the effects of architecture design between business, application, and infrastructure architecture". He describes blueprints as "a plan which describes the deployment of an architecture building block across the enterprise. It pictures the landscape of this building block in a matrix of two business dimensions".

Some examples of these blueprints (Rohloff, 2008: 564) are those in

- Application Landscapes:
  - o link to business architecture – describing for each business process how it is supported by applications;
  - o link to infrastructure architecture – describing the deployment of the application into organisational units.
- Data Repository Landscapes:
  - o link to application architecture – describing the deployment with databases and how the support defines information clusters of the information architecture;
  - o link to business architecture – describing the deployment of the databases in organisational units.
- Service Landscapes:
  - o link to infrastructure architecture – describing the deployment of infrastructure services and the support of applications;
  - o link to business architecture – describing the deployment in organisational units.

The integration strategy of Rohloff's EA framework is thus to make use of different views, which are interlinked through creation of dependency blueprints.

6.5.2.6 CEiSAR cube

The *Centre d'Excellence en Architecture d'Entreprise* (CEiSAR) investigated existing enterprise architecture frameworks/models and the practical application thereof. They developed an EA model to achieve a simplified but consistent view of the enterprise, called the 'Enterprise Architecture Cube'. This cube, with each sub-cube numbered for reference purposes, is illustrated in Figure 6-33. The three-dimensional cube depicts the complexity, the synergy and the agility dimensions. The key design principles are:

- consistency – the same set of concepts is used to describe the actors, actions and information but should not influence the presentation of different views;
- continuity – avoid discontinuity between different approaches by aligning organisation, processes and data; and
- metrics – quantify the system characteristics to report on and monitor over time (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 26).

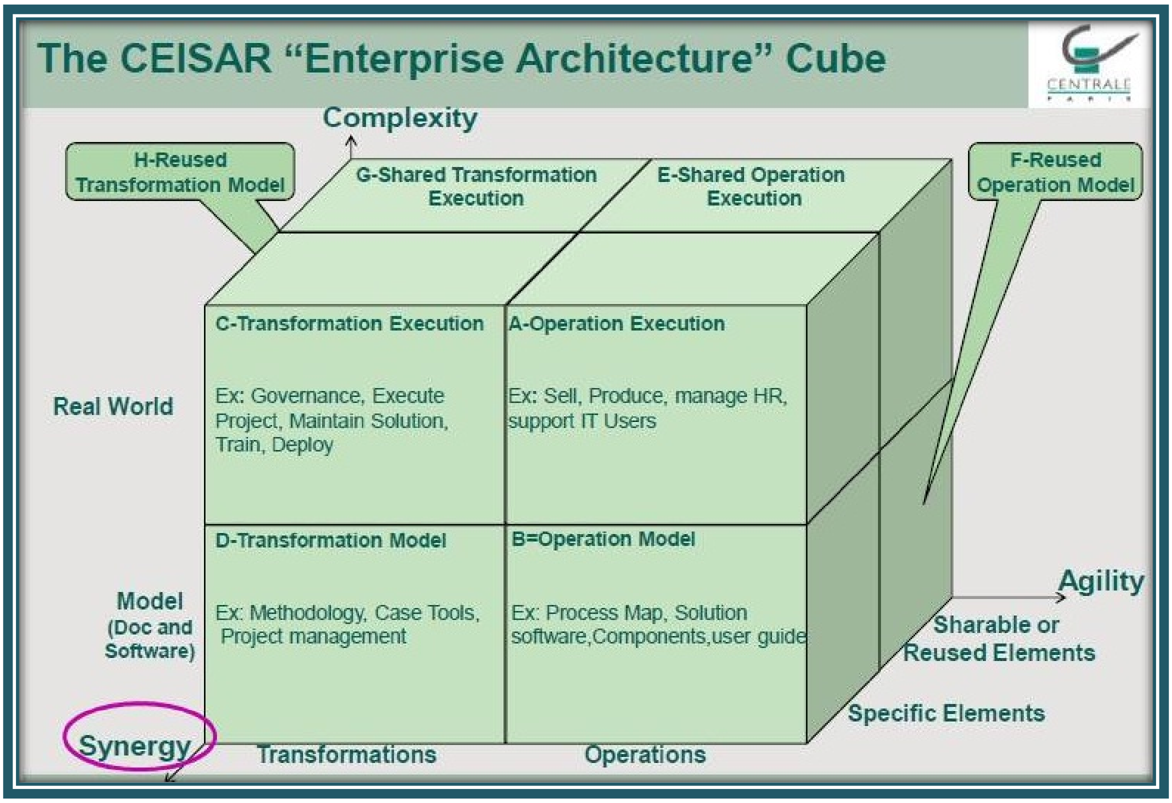


Figure 6-33: The CEiSAR Enterprise Architecture Cube (adapted from CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 30)

The **complexity** dimension splits the cube in half horizontally:

- the bottom half (sub-cubes D, B, H & F) – addresses the EA model, for example, maps, actor models, computer models, documentation, software and data model;
- the top half (sub-cubes C, A, G & E) – addresses the real world execution, for example, processes, activities, functions and real information (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 27).

The **agility** dimension splits the cube in half vertically:

- the right half (sub-cubes A, B, E & F) – addresses the operations, for example, production, service delivery, cash management and IT operation management;
- the left half (sub-cubes C, D, G & H) – addresses the transformations, for example, project execution, research and development, transformation process and defining roadmaps (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 28).

The **synergy** dimension splits the cube in half vertically from side to side:

- the front half (sub-cubes A, B, C & D) – addresses the specific elements, for example, company specific elements when working with a group of companies;
- the back half (sub-cubes (E, F, G & H) – addresses the sharable elements for the real world, for example solution units and master data, as well as reusable elements for the model, for example, actor models, process models, function models and data model (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 29).

The different dimensions give a specific focus to each sub-cube:

- Sub-cube A: Real World + Operations + Company Specific = Operation Execution.
- Sub-cube B: Model + Operations + Company Specific = Operation Model.
- Sub-cube C: Real World + Transformations + Company Specific = Transformation Execution.
- Sub-cube D: Model + Transformations + Company Specific = Transformation Model.
- Sub-cube E: Real World + Operations + Shared Elements = Shared Operation Execution.
- Sub-cube F: Model + Operations + Reused Elements = Reused Operation Model.

- Sub-cube G: Real World + Transformations + Shared Elements = Shared Transformation Execution.
- Sub-cube H: Model + Transformations + Reused Elements = Reused Transformation Model.

The integration or inter-dependencies is well defined in this model, due to its three-dimensional nature. The synergy split focuses more specifically on integration, especially with the sharable/reusable elements and the use of consistent concepts.

The integration strategy of the CEiSAR cube is thus to make use of a 3D-cube to define the inter-dependencies, which are interlinked through the synergy split process, for example, shared transformation execution.

6.5.3 Models

Models, as reasoned in paragraph 2.8, are used to fill in the different structures and sections of a framework. There are models/methods that can also be used to assist the integration of the architectural domains:

- Delineation – differentiation of information systems (Pessi, Magoulas & Hugoson, 2011: 54). Two delineation principles are discussed in Magoulas' Swedish dissertation (as cited by Pessi, Magoulas & Hugoson, 2011: 54), namely the information-driven principle and the responsibility-driven principle. Saat *et al.* (2010: 16) also use delineation as part of the metamodelling process.
- Interoperability – integration between information systems (Pessi, Magoulas & Hugoson, 2011: 55). The level of interoperability is determined by the following principles: unification principle, intersection principle and interlinking principle.
- Modularity – utilising a common modular structure to create loosely coupled entities, which are aligned and flexible (Huysmans, Ven & Verelst, 2010: 77).
- Capability mapping (SOA Consortium EA2010. Working Group, 2010: 10) – a capability encapsulates all relevant attributes and provides a stable and objective view, as it manages what service is delivered at what level and distinguishes between the 'how' and the 'what' (Doig, 2007: 15).

- Value chain analysis (SOA Consortium EA2010. Working Group, 2010: 10) – a framework encompassing the suppliers, products and services through the different architectural domains to products, services and customers based throughout on strategy (Ballengee, 2010a: 149).
- Business Engineering Navigator (BEN) – to "support construction, navigation and analysis functionalities for artifacts and relationships of all architectural layers – from strategic aspects down to IT infrastructure" (Aier *et al.*, 2009: 36).
- IBM's Component Business Model – an approach to drive a specialised focus internally and/or externally by evaluating the business components in order to expand and evolve (IBM Institute for Business Value, 2005: 5).
- Activity Model (Okon, 2010: 11) – map of activities, their interactions, input and output, resources and controls (Business Dictionary, 2010).
- Process Model (Okon, 2010: 11) – description of processes in terms of goal, inputs, outputs, resources, impact, order of activities and value.

These models, mentioned above, are methodologies which can be utilised in creating the necessary content for EA and integration artefacts.

6.6 SUMMARY AND CONCLUSION

6.6.1 Summary

The relationships between the different architectural domains were investigated. The first sub-research question answered in this chapter is: *f) What are the taxonomy and relationships of the different architectural domains?*

This question was answered by

- providing an overview of the literature – it was evident that there is a variety of different depictions of enterprise architecture;
- subsequently proposing a taxonomy – see Table 6-1; and
- proposing an integrated depiction of enterprise architecture – see Figure 6-8.

The second sub-research question answered in this chapter is: *g) What definitions, benefits and models are there for integrating, interacting and/or interlinking the* *architectural domains or are utilised for indicating the relationships between the architectural domains?*

This question was answered by

- providing an overview of the definitions in literature – in the absence of specific definitions, the different terms used in the literature were investigated and placed on a relationship continuum – see Figure 6-10. The terms 'alignment', 'relations' and 'integration' have the highest occurrence – see Figure 6-11. The terms *'relation'*, '*integration strategy'* and '*interlinking mechanisms'* will be used throughout the rest of this thesis.
- providing a synthesised definition (see Figure 6-12):

  Relating architectural domains is the integration strategies and the interlinking mechanisms used to combine two or more of business architecture, information architecture and technology architecture to provide meaningfully aligned enterprise architecture, in the context of an enterprise.
- determining the role of relating the architectural domains – clear, global, co-ordinated, unambiguous modelling to manage complexities and enable innovation;
- determining the benefits of relating the architectural domains – enabling the innovation/change and business processes, flexibility and the re-use/sharing of a variety of objects;
- investigating the relationships between the architectural domains in the main EA frameworks – see Table 6-3 below for an overview. It was found that the relations of EA domains consist of two components, being the integration strategy (what to relate) and the interlinking mechanisms (how to relate):
  - The pure Zachman framework has some integration built-in through the intersection of the building blocks in the columns and rows. The interlinking mechanisms are, however, dependent on the individual(s) documenting the EA and their insight. This can be expanded by also matching specific columns with each other (Finkelstein, 2006: 245). The Zachman DNA uses the same building blocks but adds a number of third dimensions of interlinking mechanisms, called 'sciences' (O'Rourke, Fishman & Selkow, 2003: 586). Graves (2010) also adds a third dimension, called segments, containing *inter alia* processes.

- o TOGAF establishes relation between the different architectural domains through the use of the integration strategy of an overarching metamodel (meta-architecture), including taxonomy, common language and reference architecture. The interlinking mechanisms can be enhanced by making use of ontologies (Gerber, Kotzé & Van der Merwe, 2010: 58).
- o DoDAF's integration strategy is to make use of a metamodel enabled by viewpoints, metadata and taxonomies to address the integration of the architectural domains. A net-centric common language forms part of the interlinking mechanism.
- o FEA also makes use of a metamodel strategy for integrating and re-use of shared assets like general profiles, standards, processes and the overarching Consolidated Reference Architecture. It contains transitional processes, such as investment review, segment coordination, market research and asset management, for interlinking mechanisms.
- o The integration strategy of the GEAF addresses different viewpoints which are reunited/reconciled (integrated) through the Enterprise Solution Architecture. The Gartner Enterprise Architecture Process Model provides the methodology and processes as the interlinking mechanisms.

- describing the integration strategies and interlinking mechanisms used by other frameworks – see Table 6-3 below for an overview:
  - o The integration strategy of domain architecture is to make use of finely grained viewpoints called domains, which are interlinked through partitioning – aligning the usage and the construction aspects of the relevant domains.
  - o The integration strategy of service architecture is to make use of service layers, which are interlinked through the process of building relations between the service layers, for example, 'used by', realisation, orchestration and capabilities.
  - o In viewpoints the integration strategy is to make use of particular viewpoints to determine the relevant EA components as well as to provide mechanisms to create integrations relevant to the particular viewpoint target group.
  - o In metamodels the integration strategy is to make use of a higher level of metamodel, which is interlinked through the process of using common languages, templates and reusable components.
  - o The integration strategy of Rohloff's EA framework is to make use of different views, which are interlinked through creation dependency blueprints.

- o The integration strategy of the CEiSAR cube is to make use of a 3D-cube to define the inter-dependencies, which are interlinked through the synergy split process, for example, shared transformation execution.
- listing a few examples of models/methods for creating integration content, such as delineation, interoperability, modularity, capability mapping, value chain analysis, business engineering navigator, component business model, activity model and process model.

FRAMEWORKS	INTEGRATION STRATEGY	INTERLINKING MECHANISM
EA FRAMEWORKS		
Zachman	2D-Matrix	Human intervention and alignment of columns
Zachman DNA	Multiple 3D-Matrix	Integrated activities in third dimension
Zachman (Graves)	Multiple 3D-Matrix	Built-in processes in third dimension
TOGAF	Metamodel (standard taxonomy, reference architecture)	Common Language
TOGAF (Gerber, Kotzé & Van der Merwe)	Metamodel and Ontologies	Common Language & Ontologies
DoDAF	Views, metadata and taxonomy	Net-centric common language
FEA	Metamodel (reference architecture)	Architecture processes (transitional)
GEA	Meta-architecture – solution architecture	Process model
INTEGRATION FRAMEWORKS		
Domain Architecture	Viewpoints	Partitioning (inter-domain links)
Service Orientation	Service layers	Relationships (*e.g.* used by)
Viewpoints	Viewpoints	Iterative stakeholder processes
Metamodels	Metamodel	Common language & templates
EA Framework (Rohloff)	Views	Blueprints
CEiSAR	3D-Cube	Synergy split

Table 6-3: Summary of Integration Strategies and Interlinking Mechanisms

6.6.2 Conclusion

There is a variety of depictions of EA to illustrate the relationships between the different architectural domains. The proposed integrated diagram of EA adds value as it provides a holistic view on EA, its main domains and its possible sub-domains in one depiction. This is essential for understanding, promoting and conveying EA within the enterprise and its executive. It also attempts to express the interrelations and alignment between the main domains, which aggregate in solutions architecture. This constitutes a step in the modelling process (to be addressed further on), but is not the final diagram or framework.

The terms used to define the relationships between the architectural domains, as well as the role and benefits thereof were discussed. A definition followed – see Figure 6-12.

The relation between architectural domains has been incorporated in architecture frameworks by making use of integration strategies, such as two and three dimensional matrices, metamodels, viewpoints, service layers and reference and solution architecture and by making use of interlinking mechanisms, including human intervention, common language, iterative and re-usable processes and blueprints. For example, the use of language forms an important part of the integration strategies of TOGAF (TRM), DoDAF (All View), FEA (CRA), Viewpoints and Metamodels. See Table 6-3 for an overview. Successful implementation remains a challenge to apply in practice.

This chapter concludes this section, namely *Existing Models and Frameworks for the Components of Enterprise Architecture.* The next section is *Design of an integrated Model/Framework*, which will describe the modelling process and the proposed integration framework.

SECTION B: DESIGN OF A RELATIONAL METAMODEL

The previous section, *Literature Study*, researched the current status of enterprise architecture. The research included enterprise architecture, business architecture, information architecture and technology architecture. This encompassed the history, roles, definitions, documenting, frameworks, models, ontologies and descriptive languages. The outcomes for each architectural domain are:

- synthesis of defining terms used in literature;
- proposed new synthesised definition;
- overview of existing frameworks, models, ontologies and languages.

Furthermore, the relationships between the architectural domains were researched in terms of depictions, integration strategies and interlinking mechanisms.

All the different frameworks and their inter-relationships create confusion and ambiguity for creating an effective cohesive set of enterprise architectures for an enterprise. The purpose of this section, *Design of a Relational Metamodel*, is to propose a metamodel to provide structure and order and to provide a solution to documenting the relations between the different architectural domains.

This section contains two chapters. The first, Chapter 7, will address the generic design modelling methodology to be followed to create and to document the proposed metamodel. The methodologies followed to create the existing frameworks are not clearly described in the literature in the detail required to create an own metamodel. The second, Chapter 8, will address the proposed relational metamodel, produced by applying the modelling methodology. This metamodel can be applied within enterprises to create, document and manage its enterprise architecture.

7 THE MODELLING METHODOLOGY

7.1 INTRODUCTION

Gartner (Guevara & Robertson, 2011: 1) found that a big barrier for EA lies in the inconsistent application, missing interdependencies and the inability to unify the different architectural domains. A relational metamodel is required to provide the necessary relations between the architectural domains. The term ‘relational’ is used here to indicate the relations between the different architectural domains.

The methodologies which were followed to create the existing frameworks are not described in detail in the literature. This chapter will thus address the generic design modelling methodology to be followed to create and to document the proposed metamodel.

The sub-research question to be answered in this chapter is:

i) *What will the **modelling** process look like, including the elements and deliverables which need to be addressed?*

This question will be answered by outlining the chosen modelling process, followed by the different modelling elements and the modelling deliverables.

The term metamodel will be used throughout Section B and Section C in order to avoid the ambiguity of the terms ‘framework’, ‘model’ and ‘metamodel’. According to the Open Group (2009c: 33) a metamodel is a “model that describes how and with what the architecture will be described in a structured way”. The term ‘metamodel’ will also be used when citing authors although they might have used one of the other terms.

7.2 MODELLING PROCESS

The purpose of this sub-section is not to debate or refine existing modelling processes, but to outline the process that is being followed in this designing of a relational metamodel. The agile modelling community argues that the process does not consist of steps or phases but rather of concepts, that need to be addressed, such as investigating requirements, analysis and design (Ambler, 2011). The modelling process will be discussed according to steps. The steps are, however, not necessarily followed in a waterfall approach and will entail iteration. Kruchten (1995: 13) advocates that "the architecture is actually prototyped, tested, measured, analysed, and then refined in subsequent iterations." Figure 7-1 depicts an example of an iterative approach.

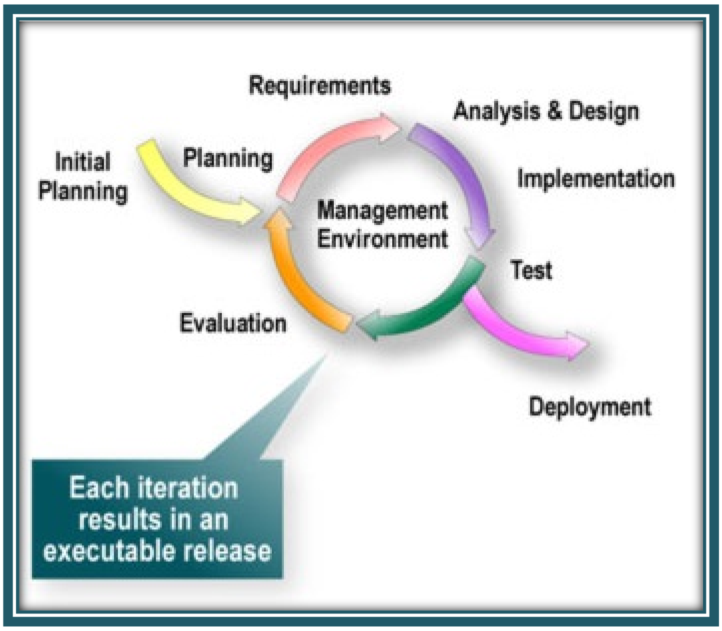


Figure 7-1: An Iterative Approach to Development (Kruchten, 2000)

The modelling process, described below, is based on a combination and selection of existing good practices from several resources (Burgstahler, 2012; Cline, *s.a.*; *The Engineering Design Process*, *s.a.*; Frazier, 2010; *Instructional Design: Using the ADDIE Model*, *s.a.*; Smit, 2008).

7.2.1 Defining the Problem

The first step is to understand, to define and to limit the problem to be addressed. In the analogy of a project and project terminology, this will constitute the project initiation phase. Defining the problem was addressed by Chapter 1 in this thesis. In Chapter 1 the problem was described and formulated as a research question. The research question was also broken down into several sub-research questions.

7.2.2 Background Research

The second step is to do orientation or background research in order to analyse the environment, to specify the requirements, to define the specifications and to provide context. Researching the background was addressed by Chapters 2 through 6 in this thesis. These chapters provided background and context by analysing and defining the different architectural domains, their role and benefits and the different existing frameworks and models. Furthermore, the relationships between these architectural domains were analysed. This was done for:

- enterprise architecture (Chapter 2);
- business architecture (Chapter 3);
- information architecture (Chapter 4); and
- technology architecture (Chapter 5).
- The inter-domain relationships (Chapter 6).

7.2.3 Design/Development

The third step is to design or develop a solution. The design principles (7.3.3) and research parameters (paragraph 1.6) as well as the orientation gained in the previous steps should be taken into account. This step involves considering alternative solutions, creating a proposed metamodel and developing it. In an engineering context, this is the production cycle of model, mock-up, prototype and product. The design, in this case the metamodel, will be documented in Chapter 8.

7.2.4 Evaluation and Refining

The fourth step in the modelling process is the evaluation, testing and refining of the metamodel – the methodology will be described in Chapter 9. External evaluation of the proposed metamodel will take place through conceptual testing in one enterprise in each of three different vertical sectors (Chapters 10, 11 and 12). The feedback from these evaluations will be used to modify and refine the metamodel (Chapter 13). Finally the applicability of the model will be tested at the University of Pretoria (Chapter 14).

Part of the evaluation is to verify that the generic design principles (paragraph 7.3.3) have been adhered to. Evaluation will make use of criteria, such as:

- an architecture scope which covers all the bases in the most simple and effective ways (Carbone, 2004: 12);
- a metamodel which has the capability to translate the architecture into a small set of well-scoped business-oriented projects (Carbone, 2004: 12);
- processes which support the architecture and the architects (Carbone, 2004: 12);
- a clear understanding of the relationships between the architectural domains;
- a standard representation scheme used throughout (Carbone, 2004: 54);
- a metamodel which can be used to define different states, for example, the current state or the target state;
- a flexible metamodel (Burgstahler, 2012: 2);
- a metamodel which is easy to apply;
- a metamodel with the correct level of simplicity and complexity (Carbone, 2004: 57);
- metamodel documentation which is clear, logical, and unambiguous; and
- a metamodel which provides a sense of the contextual perspectives when focusing on selected aspects of the enterprise (Anon., 2009).

7.2.5 Completion

The final step is the completion of the design. This could be, for example, launching the product, implementing the system or signing off the project. This step also includes communicating or marketing the result/design/product. In this case the results will be published and publicly presented.

7.3 MODELLING ELEMENTS AND DELIVERABLES

The metamodel should contain written descriptions of all its elements and deliverables. McCarthy and Menicou (2002: 339) mention the following items, which can be used in modelling:

- Governance
- Common modelling language
- Enterprise domain ontologies

- Libraries of reference models
- Knowledge representation formalisms
- An appropriate methodology.

Based on the preceding analysis of existing frameworks and models (Chapters 2 – 6), the main deliverables are discussed below. Other possible deliverables, not described here, could include: background information, architecture vision, artefact repository, assumptions, exclusions, parameters, guidelines, enterprise standards and taxonomies.

7.3.1 Applicable Standards

A component of modelling is to take into account the applicable existing standards. An example of an applicable standard is IEEE Std 1471-2000 for architectural description of software-intensive systems (IEEE, 2000). Although it describes the architecture process for a subset of information architecture, it provides basic concepts applicable throughout EA. The standard makes use of the following basic concepts:

- Architecture – "The fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution".
- View – "A representation of a whole system from the perspective of a related set of concerns".
- Viewpoint – "A specification of the conventions for constructing and using a view. A pattern or template from which to develop individual views by establishing the purposes and audience for a view and the techniques for its creation and analysis".
- Conceptual framework – the frame of reference for the architectural description.
- Context – the environment determines the setting and circumstances and other influences.
- Concerns – the interests important to one or more stakeholders.

These concepts are related and utilised by one another and are depicted in Figure 7-2 below, by using the UML Specification Class Diagram. The boxes represent classes of things and the connecting lines represent association. An association can have two roles – one in each direction. A role can have an optional label, which is placed closest to the recipient of the role, for example, *a system inhabits an environment* and *an environment influences a system.* All roles are one-to-one unless otherwise noted by *1..**. A diamond at the end of an association denotes a part-of relationship, for example, *views are a part of an architectural description.*

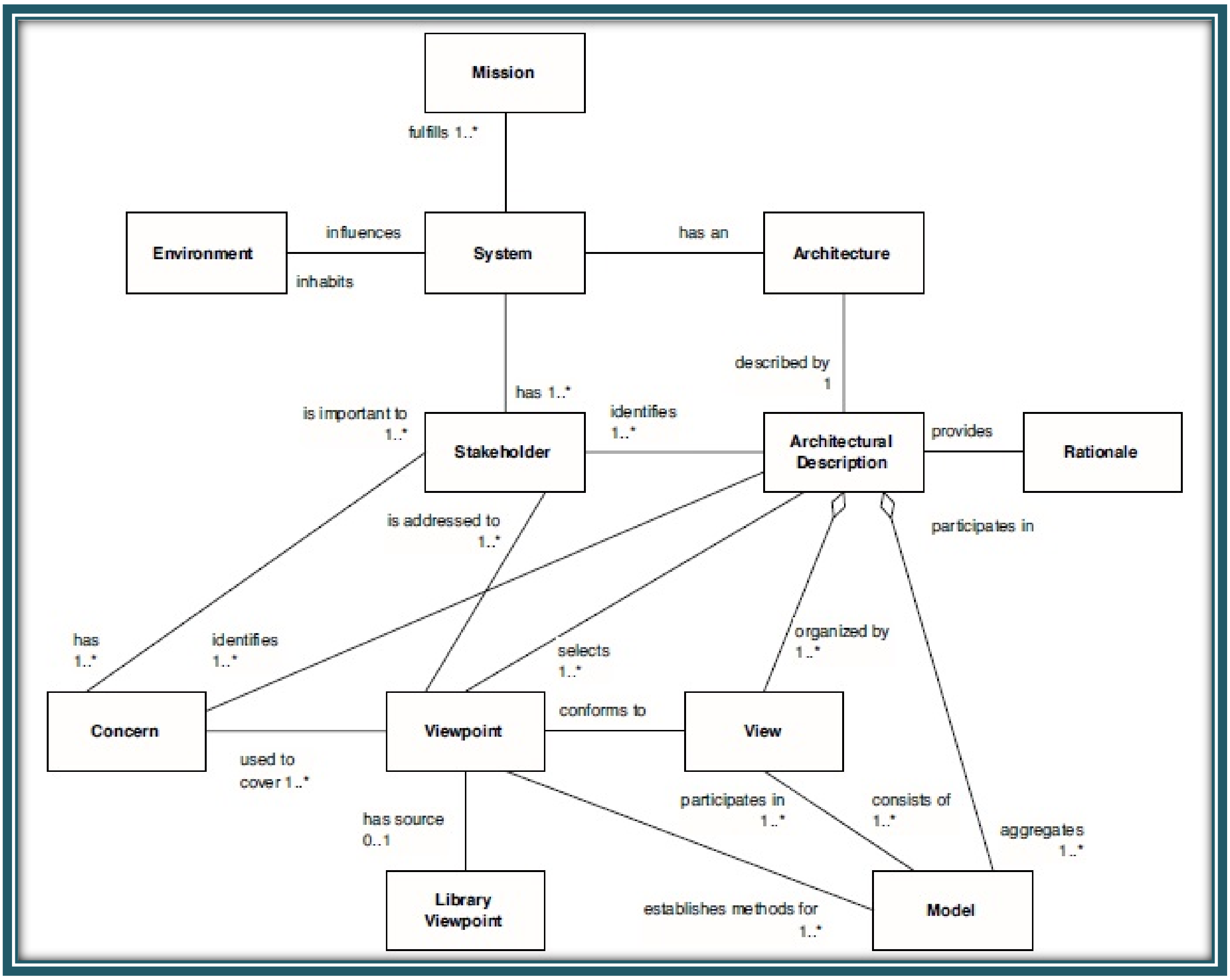


Figure 7-2: Conceptual Model of Architectural Description (IEEE, 2000: 5)

7.3.2 <u>Documentation</u>

The metamodel should set a standard for the text-based documentation created as an output of its application. When a metamodel is applied to describe an enterprise's architecture, the documentation should first contain the descriptive contents. Apart from the contents, the architectural documentation should carry at least the following metadata:

- Date of issue

- Document status
- Issuing organisation
- Change history
- Summary
- Scope
- Context
- Glossary
- References (IEEE, 2000: 8).

7.3.3 Design Principles

The metamodel should make provision for the creation and documentation of design principles as a deliverable. Developing and adhering to a set of design principles are important in order to provide guidance (Burgstahler, 2012: 2). Pessi, Magoulas & Hugoson (2011: 61) found that the principles have an impact on the responsibility of IT investments, the time to value, investment co-ordination and long-term alignment.

There are general design principles, which are applicable to the metamodel and there will be enterprise and situation-specific design principles created as a part of utilising the metamodel. The following general design principles are relevant:

- The metamodel should adhere to the relevant standards (Armenio *et al.*, 2009).
- The metamodel should be flexible to use (Burgstahler, 2012: 2) and open (Armenio *et al.*, 2009).
- The metamodel should provide the correct level of simplicity and complexity (Carbone, 2004: 57) – “how to be simple enough so that the model is understandable by all, and broad enough to cover all topics” (Carbone, 2004: 57; CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 8).
- The metamodel should be intuitive (Burgstahler, 2012: 2) – use of the design is easy to understand and follow.
- The metamodel should provide perceptible information (Burgstahler, 2012: 2) – the necessary information is communicated effectively to the user.
- The metamodel should address different levels, for example, job level, organisational unit level and the enterprise level (Anon., 2009).

- The metamodel should provide reusable components.
- The metamodel should be scalable and extensible (Armenio *et al.*, 2009).

7.3.4 Definition of Terms

The metamodel should contain clear definitions of the terms used. This has value for the usage and interpretation of the model as well as to limit misunderstandings and assumptions.

7.3.5 Depictions

The metamodel should contain a diagram or set of diagrams to depict the model in a visual mode. Visual media is ubiquitous and influence our view of the world. Few (2006: 3) reasons that the visual depiction of a model is valuable, because

- the world is predominantly experienced through the eyes;
- visual technologies are sophisticated and are becoming the norm;
- there "is no substitute for a well-designed graph when you wish to see or communicate meaningful trends, patterns, and exceptions"; and
- it displays information in a way enabling people to see clearly an accurate representation of the message.

All the researched existing frameworks and models make use of depictions (see paragraphs 2.7, 3.4, 4.4 and 5.4). In this instance, depictions could, for example, be supplied for the different levels of the metamodel, the inter-domain relationships and the taxonomy.

7.3.6 Viewpoint Specifications

Specifications of each viewpoint should be included, containing a viewpoint name, the stakeholders to be addressed by the viewpoint and the concerns to be addressed by the viewpoint. Furthermore, it should include the language, modelling techniques or analytical methods to be used in constructing a view based on the viewpoint. It could also contain a rationale for each viewpoint (IEEE, 2000: 9).

Mykityshyn & Rouse (2007: 149) provide the following diagram to explain the context of viewpoints and views – see Figure 7-3. Architecture has stakeholders who have concerns that are reflected in viewpoints. Architecture is characterised by frameworks that have views and conform to viewpoints.

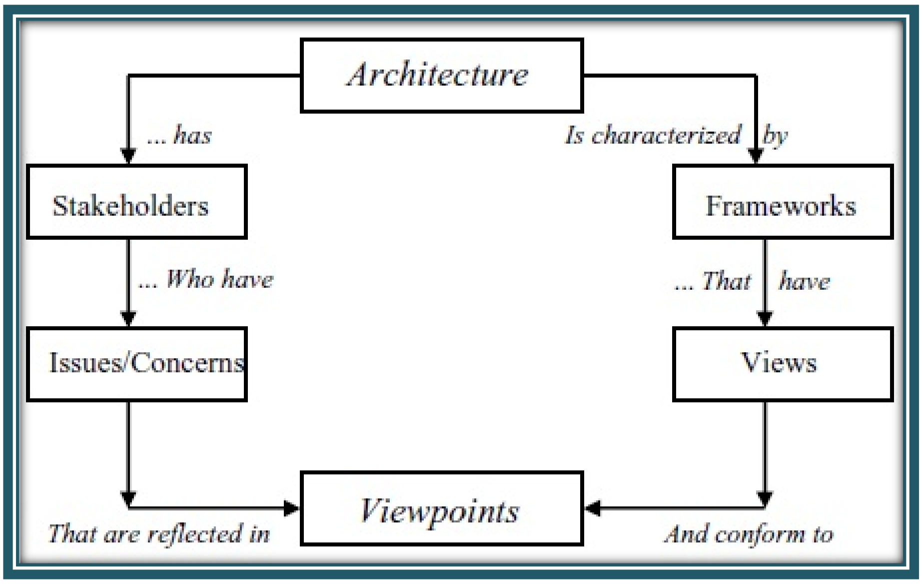


Figure 7-3: Views and Viewpoints (Mykitshyn & Rouse, 2007: 149)

7.3.7 Architectural Views

An architectural description should contain one or more architectural views. Each view should correspond to exactly one viewpoint and should conform to the specification of its corresponding viewpoint (IEEE, 2000: 10). “The idea of looking at something from a number of different points of view is not a new one and any collection of views may be thought of as an architecture” (Holt, 2009: 17). Steen *et al*. (2004) envisage a flexible approach where an enterprise (architects and/or stakeholders) can define their own views. A view can be used to view certain aspects in isolation and for relating two or more aspects.

7.3.8 Alignment Approach

The approach to aligning the different architectural domains, viewpoints and levels should be described. This could include integration strategies, interlinking mechanisms, and specific languages or notations.

7.3.9 Stakeholders and Concerns

The metamodel must be able to accommodate the identifying and documenting of stakeholders and their relevant concerns. The stakeholder list should include the role of the stakeholder (user, acquirer, developer or maintainer). The concerns should include the mission, appropriateness, risks, feasibility and maintainability (IEEE, 2000: 9).

7.3.10 Applying the Metamodel

Mechanisms for the implementation processes should form part of the architecture metamodel. The methodologies and tools were excluded from the scope (paragraph 1.6), because it was not necessary for setting the parameters of the study. However, it became evident through the study that a metamodel without an associated process is potentially a dead-end. Carbone (2004: 12) concurs with this observation by stating that unless the architecture metamodel "includes certain project-, process-, and people-focused plans and activities, even the best architecture may never begin to be realized." The initial Zachman framework is an example of a framework without the process element.

The metamodel should thus include an indication of the processes involved in creating and governing enterprise architecture for an enterprise.

7.4 SUMMARY AND CONCLUSION

7.4.1 Summary

The sub-research question answered is:

*i) What will the **modelling process** look like, including the elements and deliverables which need to be addressed?*

The first part of this sub-research question was answered by outlining the iterative modelling **process** (refer to paragraph 7.2). The process consists of:

- Defining the problem – understand, formulate and scope the problem;
- Background research – orientate, analyse environment, specify requirements, define specifications and provide context;
- Design/development – create, evaluate and consider alternatives and then model, create mock-up and/or prototype and produce the metamodel;
- Evaluation and refining – evaluate, test and refine the metamodel;
- Completion – launch the product / implement the system / sign-off the project.

The second part of this sub-research question was answered by compiling and discussing a list of modelling **elements** and **deliverables**. The metamodel should contain written descriptions of all its elements and deliverables. The resulting list of modelling elements and deliverables is as follows:

- Applicable standards
- Documentation
- Design principles
- Definition of terms
- Depictions
- Model description
- Viewpoint specifications
- Architectural views
- Alignment approach
- Stakeholders and concerns
- Methodologies.

7.4.2 Conclusion

An iterative design process will be followed to design the metamodel, including concepts such as research, development, evaluation and refining. The metamodel should include a variety of elements and deliverables in order for the applying enterprise to document their architecture effectively and efficiently.

This chapter contained a brief introduction to the modelling process which will be applied in the following chapter.

8 THE RELATIONAL METAMODEL

8.1 INTRODUCTION

This is the second chapter in the design section and will address the outcome of the modelling process, *i.e.* the creation of the relational metamodel.

The sub-research question to be answered in this chapter is:

j) How will an integrated metamodel, interlinking the different architectural domains, ***be constructed*** *and described?*

This question will be answered by following the design/development step of the design method (see 7.2.3). All the elements and deliverables, as described in paragraph 7.3, will be populated here for the proposed relational metamodel.

8.2 DESIGN/DEVELOPMENT STEP

In applying the design/development step, the significant ideas from existing frameworks will be used to provide a backdrop. A number of design alternatives will be created and discussed in order to choose a design alternative as a basis for the creation of the proposed relational metamodel.

Hybrid architecture models are prevalent. Gartner found in their 2011 EA survey (Gall, 2012: 1) a preponderance of homemade and hybrid or blended EA frameworks within enterprises. This finding “demonstrates that EA initiatives must organically create their artifacts based on the unique characteristics of their enterprises.” They also found that these hybrid frameworks are effective, implying that branded frameworks are not a requirement. Ovum’s 2012 EA survey (Blowers, 2012) concurs with this finding, as two thirds of the respondents have developed a hybrid framework. They conclude that “a framework should not be prescriptive and inflexible, but a living entity that evolves with the enterprise, retaining relevance for all stakeholders”.

8.2.1 Relevant Existing Frameworks

Existing frameworks or metamodels were analysed in Section A. There are components/ideas in these frameworks that can be incorporated or modified in the proposed relational metamodel.

First (I), the matrix of the Zachman framework has been the starting point of EA. This framework introduced the concept of different perspectives (views) and abstractions on the same information (see paragraph 6.5.1.1). The concept of views now forms part of most EA initiatives and is also described in the IEEE Standard (IEEE, 2000: 5). Graves's expansion of the Zachman Framework (Graves, 2010) adds valuable dimensions to the matrix.

Second (II), TOGAF (see paragraph 6.5.1.2) provides a very extensive and well-documented framework and process and was developed by a broad base of expertise (The Open Group, 2009a). "TOGAF provides the methods and tools for assisting in the acceptance, production, use, and maintenance of an enterprise architecture. It is based on an iterative process model supported by best practices and a re-usable set of existing architecture assets" (The Open Group, 2009c: 9).

Third (III), Gartner's concept of Solution Architecture (Guevara & Robertson, 2011: 8) forms a bridge between the different architectural domains and makes EA more practical (see paragraph 6.5.1.5).

Fourth (IV), the classification of EA viewpoints, as described by Lankhorst *et al.* (2005: 163), is valuable for differentiating between the different views in combination with the different levels of abstraction (see paragraph 6.5.2.3).

Fifth (V), the Meta-Object Facility (MOF) specification (Anon., 2012b) provides useful mechanisms for exporting models and mapping schemes (see paragraph 6.5.2.4).

Sixth (VI), the CEiSAR EA cube (see paragraph 6.5.2.6) makes use of an effective mechanism to portray the complexity of EA in a understandable but not watered down way (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 30).

There are two other applicable artefacts from Section A:

- the proposed taxonomy (see Table 6-1) (VII) and
- the proposed depiction of enterprise architecture (see Figure 6-8) (VIII).

These six above-mentioned significant ideas from existing frameworks, *i.e.* views, TOGAF processes, solution architecture, viewpoint classification, MOF processes and cube representation, as well as the other two useful artefacts, *i.e.,* taxonomy and depiction, will be taken into account to create and evaluate design alternatives.

8.2.2 <u>Considering Design Alternatives</u>

The focus of the relational metamodel, to be designed, is to define and establish the relationships between the different architectural domains. This can be described as the links, interfaces or intersections of the architectural domains. According to Pessi, Magoulas & Hugoson (2011: 55) the objective of using intersections is "to improve the availability and quality of information". There are many different types of intersections in everyday life, as depicted in Figure 8-1, which is an organised compilation of selected individual examples. Intersections can be between items of the same type (*e.g*. roads) or items of different types (*e.g*. roads and railways). Intersections can strengthen an item, for example, the intersecting support structures of a bridge. A corridor provides access to all the intersecting rooms. Some intersections base interaction on strict rules, such as border posts between countries. Intersections can also create problematic situations, for example, the leaking problem where two sections of roof meet. Another example is a traffic intersection, which can be dangerous, especially if the rules are not obeyed by everyone. Some intersections share the same resources – for example, crosswords. In spread sheets, the intersection of a row and column gives specific meaning/value to the intersecting cell. The secure intersection of a rope and a carabiner is essential for the life of a mountaineer. A walkway between two buildings provides a quick and safe intersection between them. Some intersections create new entities or groupings, for example, in a Venn diagram or a new paint colour. Intersections can be very complicated and confusing, for example, interactions in the brain, a multitude of simultaneous interactive processes and overlapping datasets.

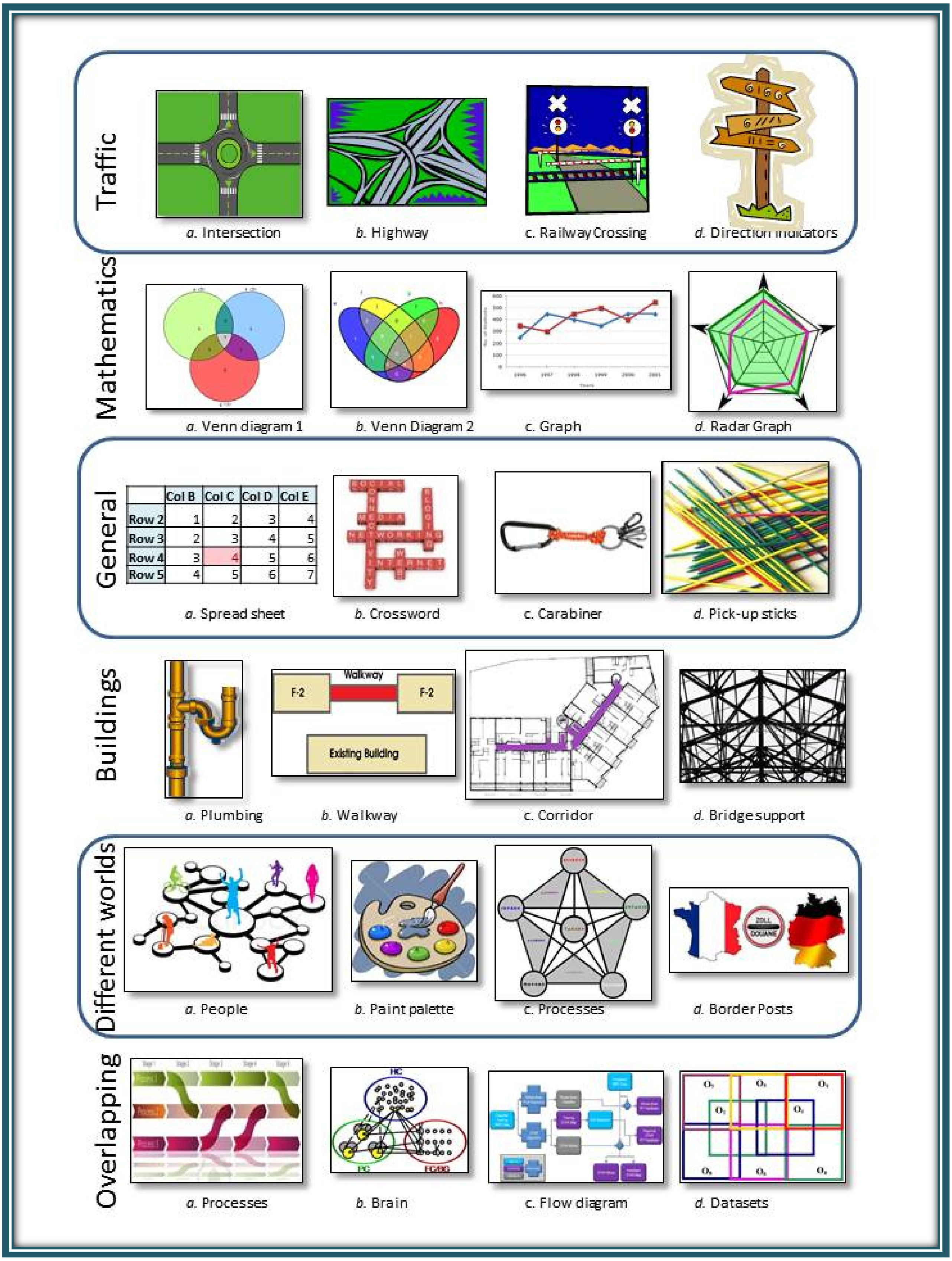


Figure 8-1: Intersection Examples

In contrast with intersections, there is the layering of the items (*e.g.* architectural domains) or a cross section thereof. Examples are a piece of multi-layered cake or a cross section of the earth's crust. The problem with presenting architectural domains in layers is that there is no depiction of the interaction or resource-sharing between the layers, as they build on top of one another. Furthermore, it is easier to depict complex relations with a three-dimensional drawing than with a two-dimensional drawing.

Three design alternatives were developed and are described below. The same colour convention, as described in Chapter 1, is used throughout for the different architectural domains.

8.2.2.1 Design Alternative A

Design Alternative A is based on the "Proposed Depiction of Enterprise Architecture" from Chapter 6, repeated here as Figure 8-2. In lieu of the iterative design methodology followed, the proposal is revisited and modifications are proposed.

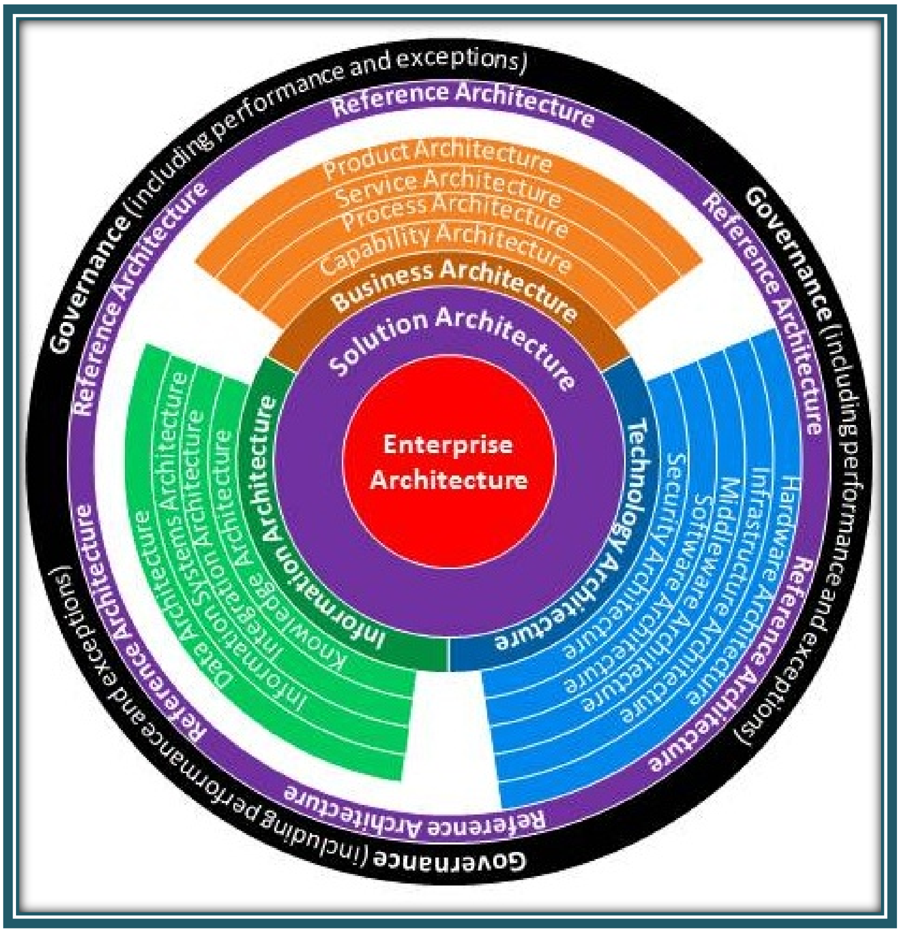


Figure 8-2: Design Alternative A

In Design Alternative A:

- Enterprise architecture is in the centre and depicts the integrated compound or the coupling of all the other architectures.
- Solution architectures combine aspects of the architectural domains into synthesised solutions for the enterprise.
- Each of the three main architectural domains touches or interacts with the others. This is a benefit of using a circular non-layered model.
- The next level of architectures is depicted below each of the main architectural domains. This provides wider context, more detail and a taxonomy.
- The reference architecture forms a circle on the outside of all the architectures. This indicates that it includes reference architectures for all the different architectures within the circle. A subset of the reference architecture will form the basis of a particular solution, thus the use of the same colour as for solution architecture.
- Governance of the architectural processes, performance and exceptions are depicted as the outside circle, as it has bearing on all the other elements.

Possible problems or questions regarding Design Alternative A are:

- Security architecture is listed under technology architecture. Is this the best or only place for security architecture?

  It can be argued that security architecture can be listed under each of the three main architectural domains. TOGAF (The Open Group, 2009c: 231) addresses security architecture in their architecture vision, business architecture, information systems architecture and technology architecture. The repetition of security architecture is problematic and will clutter the depiction. Security needs a cross-cutting mechanism.
- Is the meaning of reference architecture captured correctly?

  The other way of depicting reference architecture is by adding it to every architectural domain. This will be problematic as each sub-architecture needs to provide a reference architecture and this will clutter the depiction. These arguments lead to the conclusion that reference architecture is depicted correctly.
- Is governance depicted in the best way?

Governance could also be depicted as a square frame around the metamodel. Another possibility is to put governance as a background square to provide an underlying environment for the whole metamodel.

- Are the relations, integration strategy and interlinking mechanisms visible in this metamodel?

  The interlinking is implied by the positioning and adjacency of the three main architectural domains. It is, however, not very explicit and provides no indication of mechanisms or processes. Should this alternative be chosen, this aspect will need more attention.
- Does the metamodel provide an architectural process?

  The depiction does not provide an architectural process. It is, however, challenging to incorporate a process and a metamodel in one depiction. The best way to solve this will be to provide a separate process depiction to be used in conjunction with the current one.
- Does the metamodel accommodate the current and the future states of architecture?

  The current and future states are not visible on this level, but could possibly be accommodated on a next level of abstraction should the model be developed further.

The Design Alternative A has well-defined concepts. It provides a taxonomic context to every architectural domain. The biggest problems are the lack of a cross-cutting mechanism (for security architecture), the lack of visible integration and the lack of an implementation process.

8.2.2.2 Design Alternative B

Design Alternative B is based on the overlapping mechanism found in Venn diagrams. It is depicted below in Figure 8-3. The overlapping of architectural domains is important in the light of the focus on integration strategies and interlinking mechanisms.

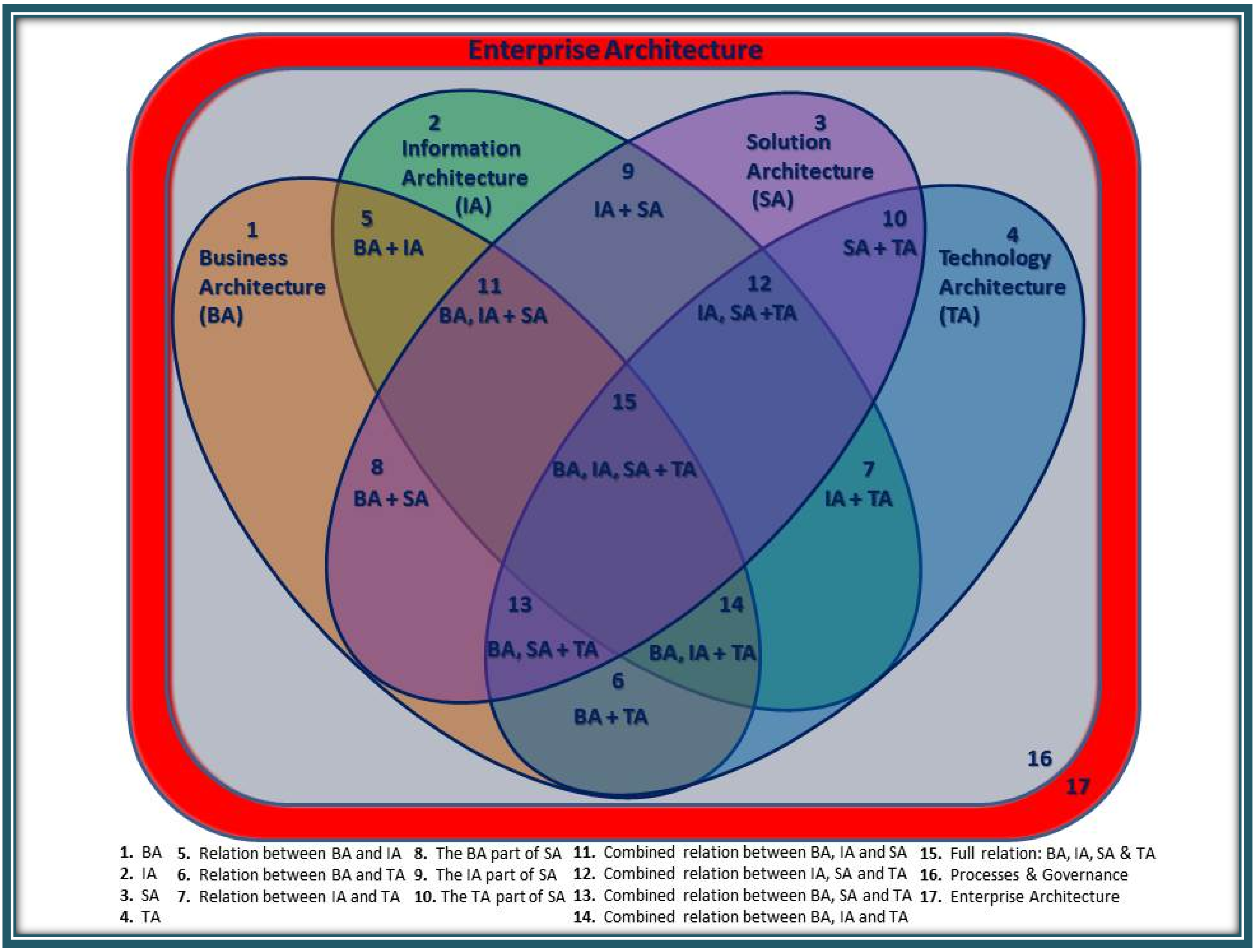


Figure 8-3: Design Alternative B

Due to the intersecting nature of Venn diagrams, this metamodel contains a list of areas. The areas are numbered for easier reference (see Figure 8-3). Each area is discussed below:

- Area 1: Business Architecture – the orange ellipse depicts the architectural domain of business architecture.
- Area 2: Information Architecture – the green ellipse depicts the architectural domain of information architecture.
- Area 3: Solution Architecture (SA) – the purple ellipse depicts the solution architecture.
- Area 4: Technology Architecture – the blue ellipse depicts the architectural domain of technology architecture.

- Area 5: Overlapping of BA and IA – depicts the parts of BA and IA which require interlinking and integration.
- Area 6: Overlapping of BA and TA – depicts the parts of BA and TA which require interlinking and integration.
- Area 7: Overlapping of IA and TA – depicts the parts of IA and TA which require interlinking and integration.
- Area 8: Overlapping of BA and SA – depicts the BA part of the solution architecture.
- Area 9: Overlapping of IA and SA – depicts the IA part of the solution architecture.
- Area 10: Overlapping of TA and SA – depicts the TA part of the solution architecture.
- Area 11: Overlapping of BA, IA and SA – depicts the combined parts of BA, IA and SA which require interlinking and integration.
- Area 12: Overlapping of IA, SA and TA – depicts the combined parts of IA, SA and TA which require interlinking and integration.
- Area 13: Overlapping of BA, SA and TA – depicts the combined parts of BA, SA and TA which require interlinking and integration.
- Area 14: Overlapping of BA, IA and TA – depicts the combined parts of BA, IA and TA which require interlinking and integration.
- Area 15: Intersection of all four ellipses – depicts the integrated BA, IA and TA part of the solution.
- Area 16: Processes and Governance – depicts the underlying process and governance of the enterprise architecture process in an enterprise. (Note: this area should be black according to the convention, but was changed to grey to increase visibility.)
- Area 17: Enterprise Architecture – the red rectangular shape depicts enterprise architecture which includes all of the above.

Possible problems or questions regarding Design Alternative B are:

- Where does reference architecture fit into the model?
  Reference architecture is not addressed *per se.* It should form part of every area.
- Is the numbering of the areas effective?
  The numbers are necessary to facilitate referring to and describing them. The order of the numbering could be modified, for example, to give EA the number ‘1’ or to number the areas from left to right sequentially and not logically.

- What is the significance of the size of the areas?
  The size of the areas is a result of the nature of the diagram and has no meaning in terms of significance or size of content. The size of the content will differ from enterprise to enterprise or from project to project. This could be misleading.
- Is Solution Architecture an architectural domain such as BA, IA and TA?
  Although the depiction implies this, solution architecture is not an architectural domain such as the other three. It is, however, valuable to depict the relations to the other architectural domains in such a manner.
- Does the metamodel accommodate the current and the future states of architecture?
  The current and future states are not visible on this level, but could possibly be accommodated on a next level of abstraction should the model be developed further.
- Does the metamodel accommodate security architecture?
  The security component of each architectural domain is not visible on this level, but could possibly be accommodated on a next level of abstraction should the model be developed further.

The Design Alternative B is effective in depicting the integration and interlinking, solution architecture as well as processes and governance. However, it does not address reference architecture *per* se and the size of the areas could be misleading.

8.2.2.3 Design Alternative C

Design Alternative C is a modified CEiSAR cube (CEiSAR. Center for Excellence in Enterprise Architecture, 2008: 30) and is depicted in Figure 8-4, below. The axes and some of the aspects have been modified. The cube consists of eight cubes combined into one. Every cube has a number (A - H) for easy reference. The compound cube can be sliced in three directions:

- X-axis (business architecture) sliced into a left and a right half:
  - left half (dark orange) indicating the operational (business-as-usual part) or current state (C, D, G and H), and
  - right half (light orange) indicating the transformational part or future state (A, B, E, and F).
- Y-axis (technology architecture) sliced into a bottom and a top half:

- o bottom half (dark blue) indicating the standardised managed part (B, D, F and H), and
- o top half (light blue) indicating the agile and flexible part (A, C, E and G).

- Z-axis (information architecture) sliced into a front and a back half:
  - o back half (dark green) indicating the shared and reusable resources (E, F, G, and H), and
  - o front half (light green) indicating the specific unique resources (A, B, C and D).

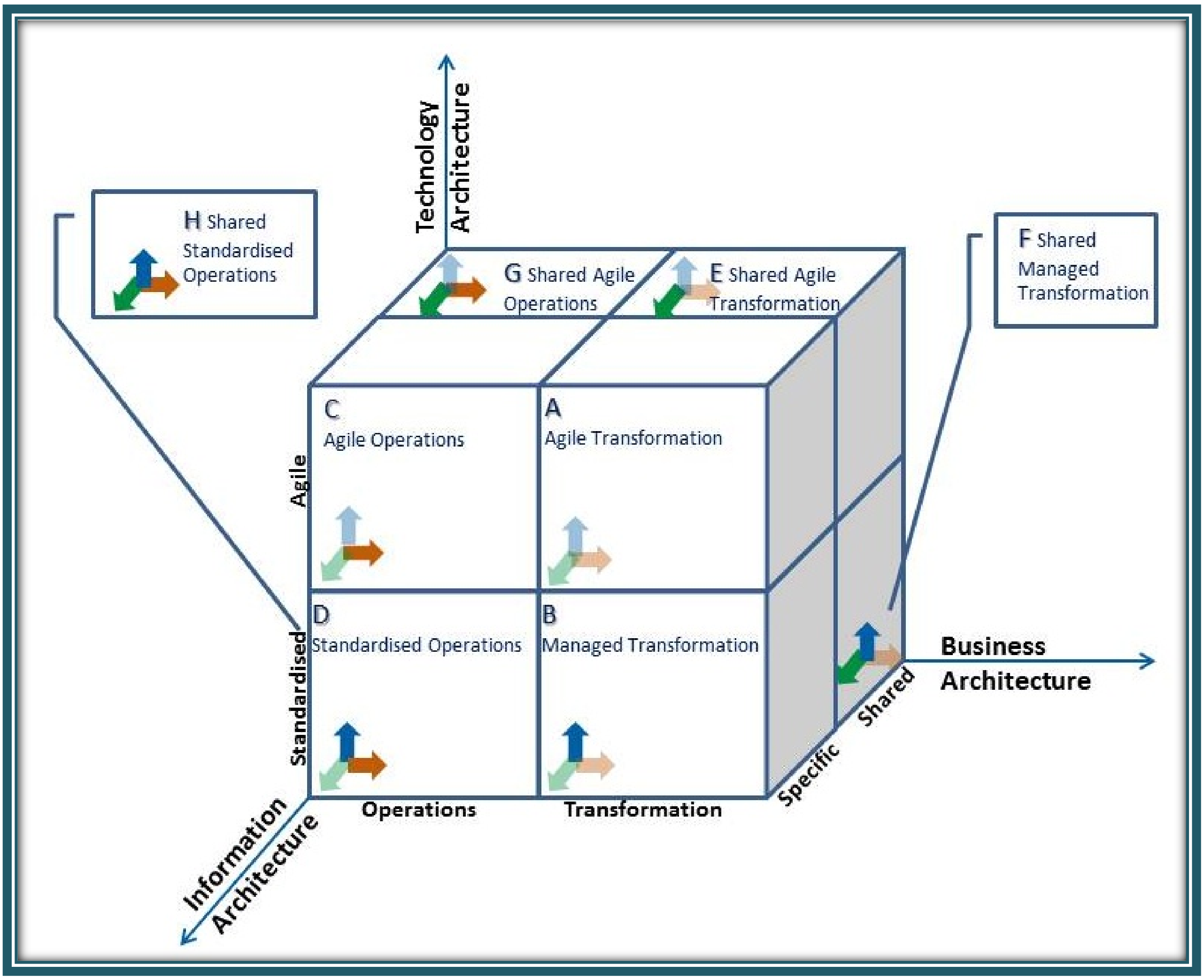


Figure 8-4: Design Alternative C

Every cube is the intersection of three aspects of the different architectural domains. Each cube is described below:

- Cube A depicts the intersection between transformational BA, specific IA and agile TA = Agile Transformation.

- Cube B depicts the intersection between transformational BA, specific IA and standardised TA = Managed Transformation.
- Cube C depicts the intersection between operational BA, specific IA and agile TA = Agile Operations.
- Cube D depicts the intersection between operational BA, specific IA and standardised TA = Standardised Operations.
- Cube E depicts the intersection between transformational BA, shared IA and agile TA = Shared Agile Transformation.
- Cube F depicts the intersection between transformational BA, shared IA and standardised TA = Shared Managed Transformation.
- Cube G depicts the intersection between operational BA, shared IA and agile TA = Shared Agile Operations.
- Cube H depicts the intersection between operational BA, shared IA and standardised TA = Shared Standardised Operations.

Possible problems or questions regarding Design Alternative C are:

- Where does reference architecture fit into the model?
  Reference architecture is not visible on this level, but could possibly be accommodated on a next level of abstraction should the model be developed further.
- Does the metamodel provide an architectural and/or governance process?
  Architectural and governance processes are not reflected in this design alternative. This is important as EA is a process and it is problematic to implement a model without guiding processes.
- Does the metamodel depict the different architectural domains effectively?
  The architectural domains form the basis of the design on the three axes.
- Where does solution architecture fit into the metamodel?
  Solution architecture is not visible *per se.* The different combinations in the different cubes provide guidance for solutions.
- Does the metamodel accommodate the current and the future states of architecture?
  The current and future states are clearly visible on this level of abstraction on the X-axis (operations – transformation).
- Does the metamodel accommodate security architecture?

The security component of each architectural domain is not visible on this level, but could possibly be accommodated on a next level of abstraction should the model be developed further.

The Design Alternative C effectively depicts the architectural domains in the current and future state. It addresses the issues of interlinking and integration by combining the different axes in each cube. Some of the drawbacks of this design are the absence of processes and the lack of visibility of reference and solution architecture.

8.2.3 Development of the Chosen Alternative

Three design alternatives have been created, discussed and considered. Design Alternative A had problems with cross-cutting mechanisms, the visibility of integration and the lack of an implementation process. Design Alternative B had problems with depicting reference architecture and security architecture. Design Alternative C had problems with reference architecture and solution architecture as well as the lack of an implementation process. It was thus decided to combine the applicable elements from all three design alternatives to create a new proposed metamodel, namely:

- The circular layout of Design Alternative A, resulting in all three main architectural domains being adjacent to each other.
- The different overlapping areas were taken from Design Alternative B.
- The three-dimensionality from Design Alternative C was utilised.

Elements based on or confirmed by existing frameworks were used as design inputs. The relevant existing frameworks were listed and numbered, from I to VIII in paragraph 8.2.1. The following elements were used:

- I – matrix and views;
- II – ADM processes and principles;
- III – solution architecture;
- IV – different levels of viewpoints;
- V – alignment language;
- VI – Design Alternative C and three-dimensionality;
- VII – Design Alternative A; and
- VIII – Design Alternative A.

The proposed metamodel will be referred to as the Relational Enterprise Architecture Metamodel or REAM. The REAM is based on a three-dimensional hexagon or hexagonal prism. A hexagonal prism has eight faces, 18 edges and 12 corners/vertices. Figure 8-5 contains an example of a hexagonal prism net. The geometrical net allows easier visibility of all sides and will be used throughout.

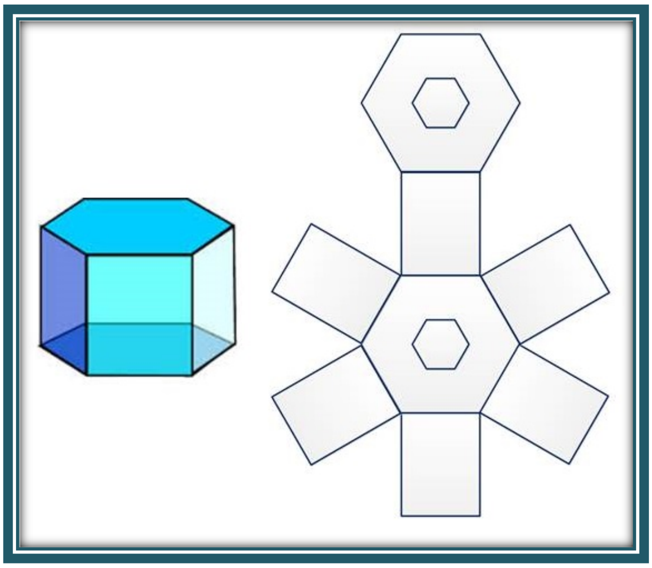


Figure 8-5: Hexagonal Prism Net

At the first level of abstraction, the **architectural domains** are displayed. See Figure 8-6. Each architectural domain is depicted in its colour and spans over a third of the top hexagon, a third of the rectangular sides and a third of the bottom hexagon. Different shades of colour are used to indicate different levels of detail. For example, dark orange (on top hexagon) indicates the overview level of business architecture; medium orange (sides of hexagonal prism) indicates the coherence level of business architecture; and light orange (on bottom hexagon) indicates the detail level and reference architecture of business architecture. The different levels are adapted from Lankhorst *et al.* (2005: 163). The levels could also be described as macro-, meso- and micro-levels. Dopfer, Foster & Potts (2004) describe it as follows: “In our view, a meso is a thing (a rule and its population) that is made of complex other things (micro) and is an element in higher order things (macro)”. The smaller black hexagon on the top indicates the architectural processes and governance and is applicable to all the architectural domains. The smaller purple hexagon on the bottom indicates solution architecture, as coined by Gartner (Guevara & Robertson, 2011: 8). Both ‘Processes & Governance’ and ‘Solution Architecture’ has bearing on all the architectural domains, although the model might imply that each forms only half of the core. Applicable security architecture should be addressed in each area.

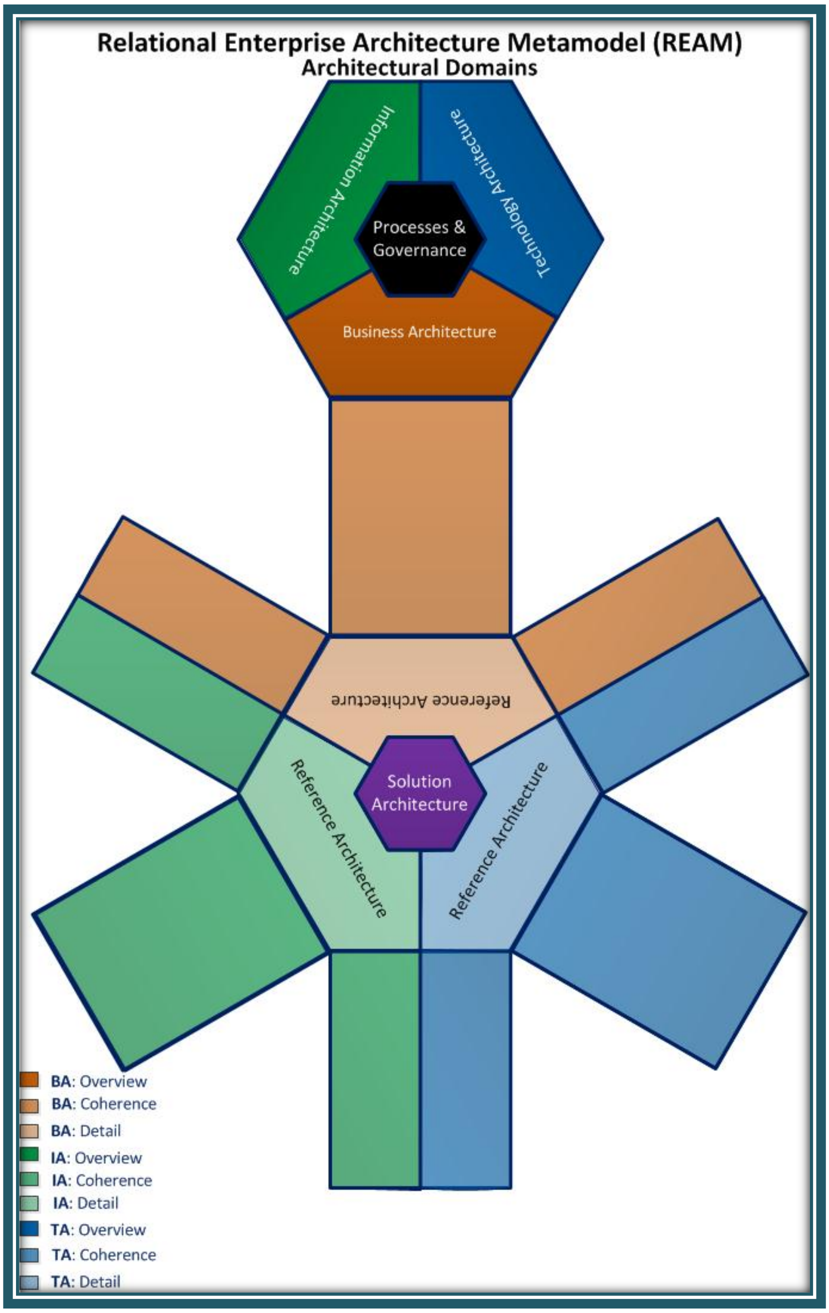


Figure 8-6: Proposed Relational EA Metamodel (REAM) – Architectural Domains

The next level of abstraction addresses the current and future-**state** architectures (see Figure 8-7). The current state indicates the baseline or “as-is” architecture. The future state indicates the target or “to-be” architecture. The middle section of each architectural domain is vertically divided in half. The left half indicates the current state architecture and the right half the future-state architecture.

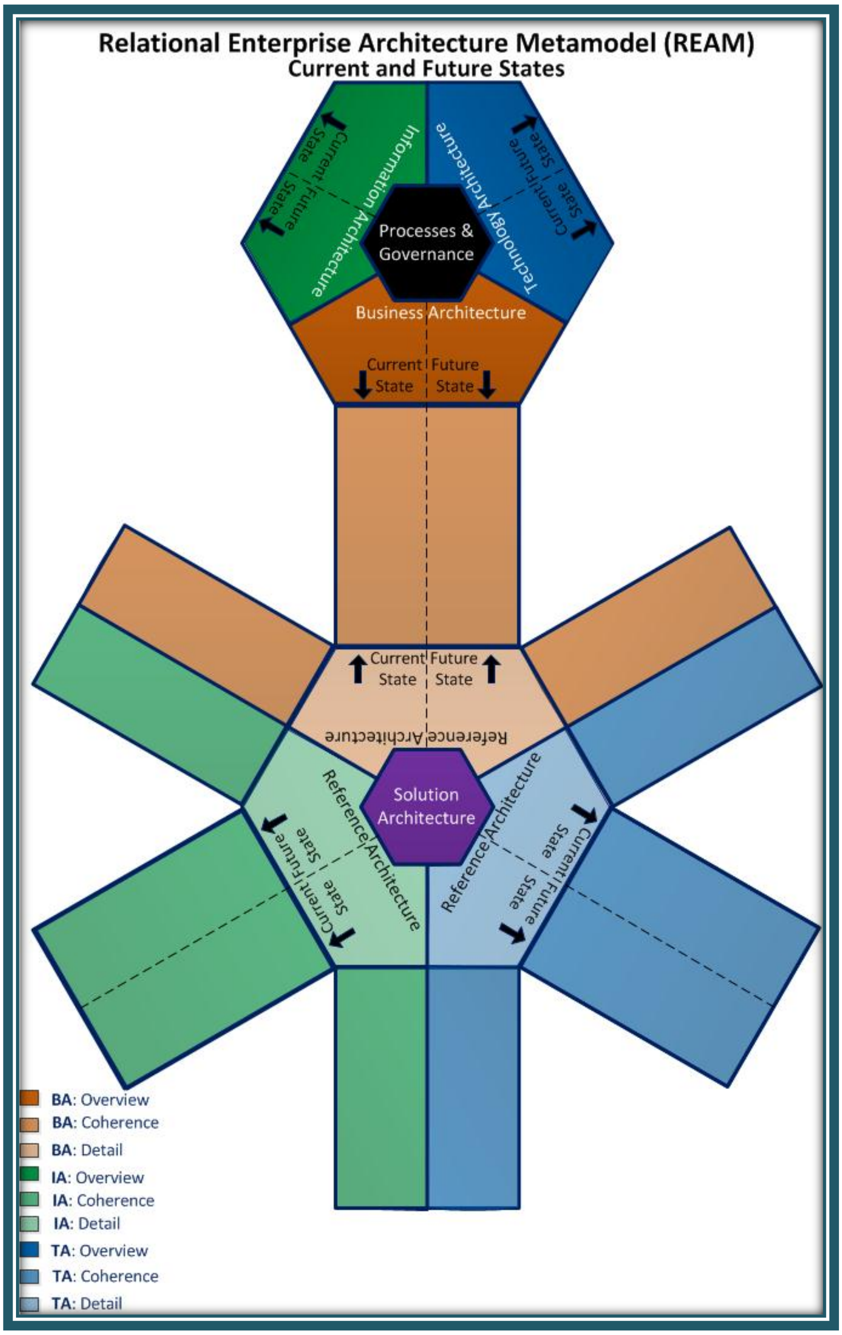


Figure 8-7: Proposed Relational EA Metamodel (REAM) with Current and Future States added

The next level of abstraction offers the capability to address different architectural **views**, see Figure 8-8. Detailed examples of views are provided in the last paragraph of section 2.1.1. The views are accommodated as horizontal bands across the rectangular sides of the model. For the time being, the views are numbered as A, B and C, but the names and the numbers of these views can be adapted as necessary. A single view spans across all the architectural domains. If, for example, security requires prominent attention in an enterprise, security could be one of the views.

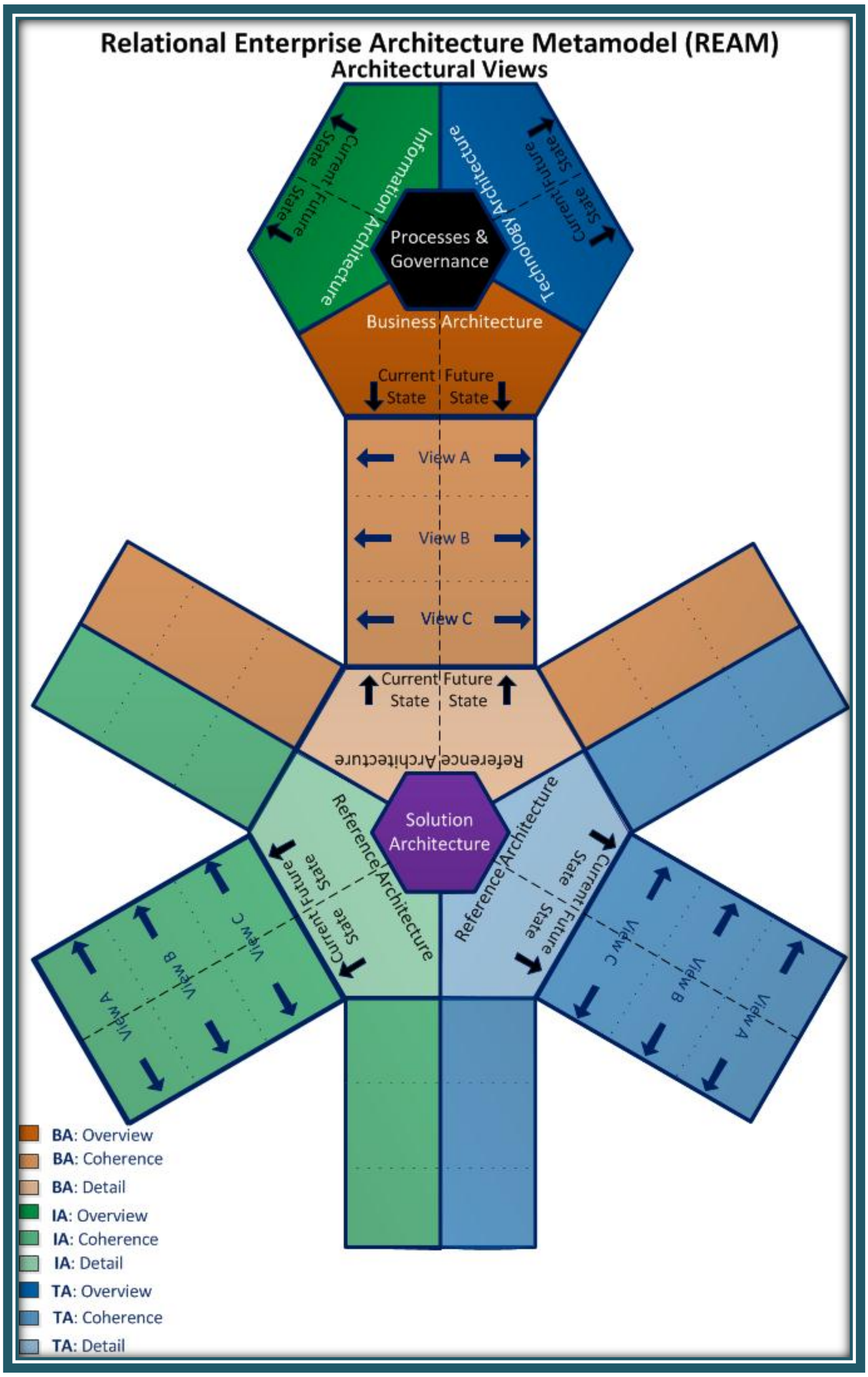


Figure 8-8: Proposed Relational EA Metamodel (REAM) with Views added

The last level of the proposed Relational EA Metamodel adds the **relations**, which are the integration strategies and the interlinking mechanisms. These are represented, in Figure 8-9, by interlinking chains through the adjacent sides of the architectural domains.

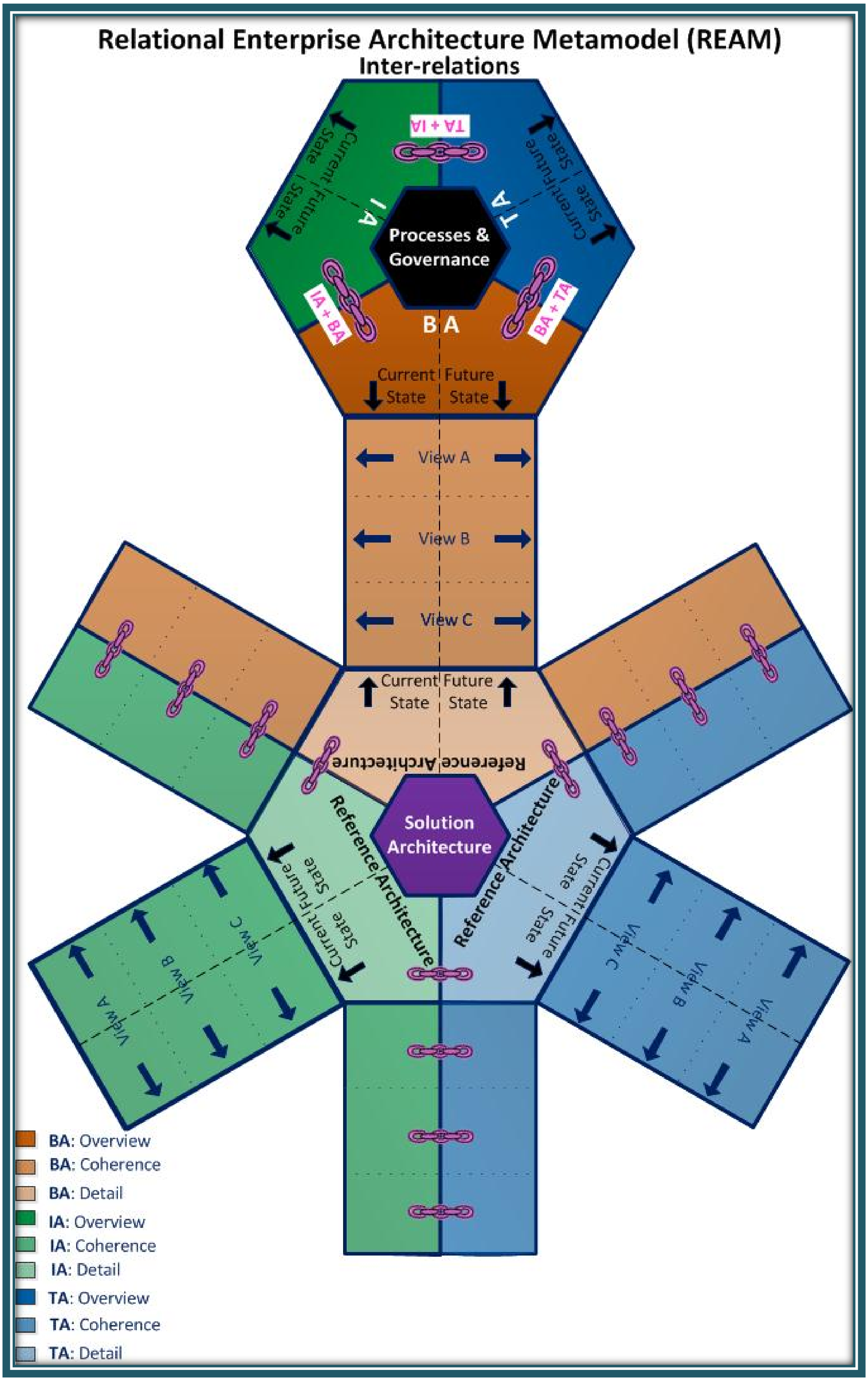


Figure 8-9: Proposed Relational EA Metamodel (REAM) with Relations added

The detail of the proposed REAM will be developed and described in the following paragraphs, according to the headings set out in Chapter 7, paragraph 7.3.

8.3 DESCRIPTION OF MODEL/FRAMEWORK

Each area in the proposed REAM is numbered for easier reference in the metamodel description. The numbered areas are displayed in Figure 8-10. P1 indicates the “Processes & Governance”-area. All the “Business Architecture”-areas are numbered B1 – B20. All the “Information Architecture”-areas are numbered I1 – I20. All the “Technology Architecture”-areas are numbered T1 – T20. S1 indicates “Solution Architecture”.

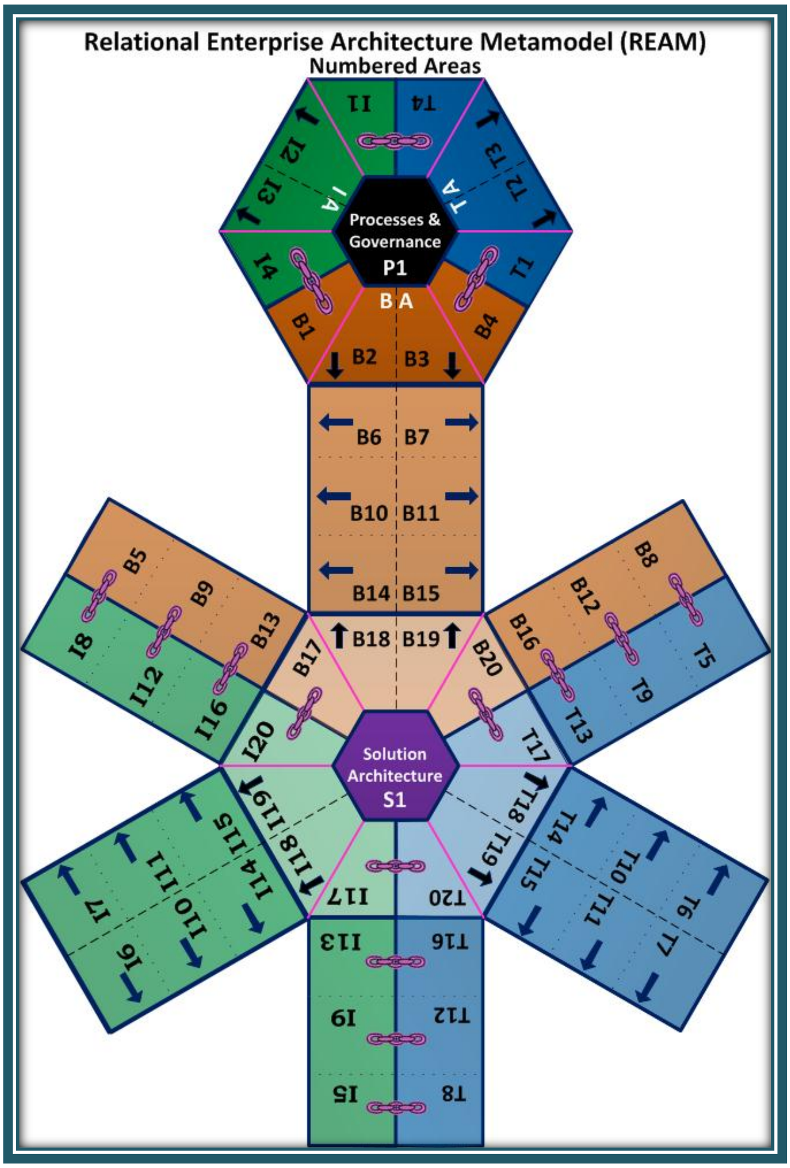


Figure 8-10: Proposed REAM with Numbered Areas

8.3.1 <u>Applicable Standards</u>

Minoli (2008: 122) indicates ISO (International Standard Organisation) 15704: "Requirements for enterprise-reference architectures and methodologies" as one of the more inclusive available standards. It attempts to place the architecture concept used in existing frameworks in an encompassing conceptual framework. "The conceptual framework is textual and relatively informal" (Minoli, 2008: 125).

Martin, Walker & Robertson (2007) state that

- EA is about enterprise project structure; and
- the standard provides
  - a description of GERAM (Generalised Enterprise Reference Framework and Architecture) as a compliant approach (see paragraph 2.7.5.3);
  - different processes;
  - life cycles;
  - stakeholder concerns;
  - viewpoints and views;
  - human aspects;
  - business mission focused;
  - genericity;
  - methodologies; and
  - modelling languages

The IEEE 1471-200: "Recommended Practice for Architectural Description of Software-Intensive Systems" (IEEE, 2000) is also an applicable standard. It contains a conceptual framework for architectural description according to concepts. These concepts were used, among others, to determine the description elements in Chapter 7 (paragraph 7.3) and are incorporated in the description of the proposed Relational EA Metamodel in the next paragraphs.

8.3.2 Documentation

During the EA process deliverables, artefacts and building blocks are created, for example, reference model, standard, architecture snapshot, network diagram, server specification, business interaction matrix, *etc.* (The Open Group, 2009c: 13). This documentation should be managed in an architecture repository. The repository should be structured according to the proposed REAM and re-usable or shareable artefacts should be easily accessible. All architecture artefacts should contain a set of metadata (see paragraph 7.3.2) for ease of reference, retrieval and coherence. An example document template is shown in Table 8-1. Each artefact/document in the repository should contain life cycle information, such as revision frequency and date.

Bibliographic Data	
Title	
Author(s)	
Date of Issue	
Organisation	
Document Data	
Number	
Version	
Status	*e.g.* draft, approved, pending, archived, *etc.*
URL	
Change History	*e.g.* previous revisions and main changes
Summary	synopsis of content
Next Revision Date	for life cycle management
Distribution Level	*e.g.* confidentiality, internally, externally, *etc.*
Architectural Data	
Area Number	REAM area, *e.g.* B2
Area Name	REAM area, e.g. Overview of BA Current State
Document Type	*e.g.* roadmap, activity model, organisation chart, *etc.*
Dependent on	other related documents in repository
Input to	other related documents in repository
Keywords	
Content	
Content	The form of the content can vary substantially, depending on the document type
Back Matter	
Glossary	
Abbreviations	
References	

Table 8-1: EA Artefact Template

Each area (see Figure 8-10) has a combination of meanings, facets and possible artefacts. Areas P1 and S1 are explained below. The detail of the other areas are summarised in a table – see Table 8-2. The table provides the meaning, example facets and example artefacts for each area. One view (View A) is listed in the table as an example. Multiple views can be added to fit the enterprise's requirements.

P1 – Architecture processes and governance:

- Processes:

  - Architecting is a part of the whole life cycle of a technology or a business process and is not a single activity at one point in a life cycle (IEEE, 2000: 6). In some scenarios the architecture can be fixed and prescriptive, in others it can coevolve with the process of creating a new system/business process or even be reversed engineered for existing systems. The life cycle of architecting should be documented as part of the governance documentation.
  - Architecture change processes
  - Requirements management, including requirements, constants, gaps and assumptions
  - Architecture vision, including business and technology strategies, drivers and stakeholders (stakeholder map matrix, value chain diagram, solution concept diagram)
  - Context of the EA
- Governance:
  - Governance process and alignment description
  - Description of governance bodies, for example, the Architecture Board
  - Architecture capability and skills framework, including architectural roles
  - Exception process and log
  - Architecture oversight of implementation and compliance
  - Architecture principles (principles catalogue)
  - Architecture maturity models

S1 – Solution architecture:

- Combined extraction/synthesis from B17, B18, B19, B20, I17, I18, I19, I20, T17, T18, T19 and T20 = detailed solution
- Portfolio charters, including capabilities, work packages and architecture contracts
- Implementation and delivery planning
- Solution characteristics (content, structure, function; and time period as well as maturity/volatility)
- Solution patterns
- Stakeholders and concerns
- Project Start Architecture (Teeuwen *et al.*, 2010: 1)

	Meta-level (overview)			
BA	**B1** *BA & IA relation* ? Overview of BA's relation with IA ❖ Stakeholders, security, information ▤ Bus info diagram	**B2** *BA current state* ? Overview of current state of BA ❖ Stakeholders, security, organisation ▤ Process flow diagr	**B3** *BA future state* ? Overview of future state of BA ❖ Stakeholders, security, migration ▤ Arch requirem spec	**B4** *BA & TA relation* ? Overview of BA's relation with TA ❖ Stakeholders, security, technology ▤ Bus footprint diagr
IA	**I1** *IA & TA relation* ? Overview of IA's relation with TA ❖ Stakeholders, security, technology ▤ System matrix	**I2** *IA current state* ? Overview of current state of IA ❖ Stakeholders, security, systems ▤ Appl portfolio cat	**I3** *IA future state* ? Overview of future state of IA ❖ Stakeholders, security, expansion ▤ Appl migration diag	**I4** *IA & BA relation* ? Overview of IA's relation with BA ❖ Stakeholders, security, business ▤ Interface catalogue
TA	**T1** *TA & BA relation* ? Overview of TA's relation with BA ❖ Stakeholders, security, business ▤ Tech portfolio cat	**T2** *TA current state* ? Overview of current state of TA ❖ Stakeholders, security, standards ▤ Tech standards cat	**T3** *TA future state* ? Overview of future state of TA ❖ Stakeholders, security, positioning ▤ Roadmap	**T4** *TA & IA relation* ? Overview of TA's relation with IA ❖ Stakeholders, security, systems ▤ Syst/tech matrix

? meaning

❖ example facets

▤ example artefact

	Meso-level (for example view A)			
View A's:	**B5** *BA & IA portion* ? The BA of View A's relation with IA ❖ Stakeholders, security, viewpoint spec ▤ Role catalogue	**B6** *BA current state* ? Current BA of View A ❖ Stakeholders, security, standards ▤ Bus functional cat.	**B7** *BA future state* ? Future BA of View A ❖ Stakeholders, security, innovation ▤ Driver catalogue	**B8** *BA & TA portion* ? The BA of View A's relation with TA ❖ Stakeholders, security, viewpoint spec ▤ Business use case
View A's:	**I5** *IA & TA portion* ? The IA of View A's relation with TA ❖ Stakeholders, security, technology ▤ Data security diagr	**I6** *IA current state* ? Current IA of View A ❖ Stakeholders, security, viewpoint spec ▤ Class diagram	**I7** *IA future state* ? Future IA of View ❖ Stakeholders, security, development ▤ Syst realisation diag	**I8** *IA & BA portion* ? The IA of View of A's relation with BA ❖ Stakeholders, security, business ▤ System use case
View A's:	**T5** *TA & BA portion* ? The TA View of A's relation with BA ❖ Stakeholders, security, business ▤ Locations diagr	**T6** *TA current state* ? Current TA of View A ❖ Stakeholders, security, viewpoint spec ▤ Platform decompos.	**T7** *TA future state* ? Future TA of View A ❖ Stakeholders, security, environment ▤ Tech trends o/view	**T8** *TA & IA portion* ? The TA of View of A's relation with IA ❖ Stakeholders, security, applications ▤ Processing diagr

	Micro-level (reference architecture)			
Reference Architecture of:	**B17** *BA & IA relation* ? Reference architecture of BA & IA ❖ Stakeholders, security, systems ▤ Event diagram	**B18** *BA current state* ? Reference architecture of current BA ❖ Stakeholders, security, detail ▤ F. Decompos. diagr	**B19** *BA future state* ? Reference architecture of future BA ❖ Stakeholders, security, detail planning ▤ Business use case	**B20** *BA & TA relation* ? Reference architecture of BA & TA ❖ Stakeholders, security, technology ▤ Prod lifecycle diagr
Reference Architecture of:	**I17** *IA & TA relation* ? Reference architecture of IA & TA ❖ Stakeholders, security, detail ▤ Appl interaction matr	**I18** *IA current state* ? Reference architecture of current IA ❖ Stakeholders, security, detail ▤ Data dissemination	**I19** *IA future state* ? Reference architecture of future IA ❖ Stakeholders, security, migration ▤ Data lifecycle diagr	**I20** *IA & BA relation* ? Reference architecture of IA & BA ❖ Stakeholders, security, business ▤ System use case
Reference Architecture of:	**T17** *TA & BA relation* ? Reference architecture of TA&BA ❖ Stakeholders, security, performance ▤ IT/Bus services matr	**T18** *TA current state* ? Reference architecture of current TA ❖ Stakeholders, security, production ▤ Network topology	**T19** *TA future state* ? Reference architecture of future TA ❖ Stakeholders, security, implementation ▤ Detailed roadmaps	**T20** *TA & IA relation* ? Reference architecture of TA & IA ❖ Stakeholders, security, detail ▤ System/tech matrix

Table 8-2: Description of Areas of the REAM

8.3.3 Architecture Principles

Principles are:

- "general rules and guidelines, intended to be enduring and seldom amended" (The Open Group, 2009c: 265);
- "the basic and most important reasons for doing or believing something" (Cambridge Dictionaries Online, 2011);
- "a fundamental truth or proposition that serves as the foundation for a system of belief or behaviour or for a chain of reasoning" (Oxford Dictionaries, 2011).

Principles can be established on multiple levels, for example, enterprise level, IT level and architecture level. The focus will be on architecture principles. Architecture principles provide the rules/reasons/foundation for the use and deployment of all IT resources and assets across the enterprise. These principles portray a level of consensus within the enterprise and provide the basis for future IT decisions. "Each architecture principle should be clearly related back to the business objectives and key architecture drivers" (The Open Group, 2009c: 266). Shah & El Kourdi (2007: 36) view principles as a way to enhance the alignment within the enterprise. Cascading principles could probably assist with the alignment.

According to IBM (Schultz, 2007) the attributes of a good principle are: simplicity, consistency of interpretation, relevancy, granularity, flexibility and stability. The principles should be documented in a consistent manner. Based on IBM's (Schultz, 2007) and TOGAF's (The Open Group, 2009c: 266) conventions, the following fields are proposed in Table 8-3.

FIELD	DESCRIPTION
Name/Number	An easy meaningful name capturing the essence of the principle and a number for reference purposes
Statement	Succinct and unambiguous fundamental rule
Rationale	Business benefits, intention of principle and relationships with other principles
Implications	Requirements for business and IT for carrying out the principle (*e.g.* resources, costs, activities).

Table 8-3: Format for Principles

Principles can be developed through the architecture process. In case the number of principles increases unmanageably, they should be grouped together in groups. In order to fit the REAM, the following groupings are suggested:

- Governance & Management
- Business
- Information/Data
- Technology
- Solutions.

Every enterprise should compile its own applicable list of principles. The following overviews are examples of architecture principles, within groups, without supplying the detailed template for each principle. These principles are a regrouped and rephrased combination of example principles from IBM (Schultz, 2007) and TOGAF (The Open Group, 2009c: 269). The order of the principles within the group is not significant. Table 8-4 contains the example principles for Process and Governance. Table 8-5 contains the example principles for Business Architecture, Table 8-6 for Information Architecture and Table 8-7 for Technology Architecture. The sample Solution Architecture principles also include input from Guevara & Robertson (2011: 9) and are displayed in Table 8-8.

Process, Governance and Management Principles (PP):	
PP1	**Primacy of Principles** These principles apply to all enterprises/units within the enterprise.
PP2	**Compliance with Law** Enterprise information management processes comply with all relevant laws, policies and regulations.
PP3	**IT Responsibility** The IT department/unit is responsible for owning and implementing IT processes and infrastructure which enable solutions to meet user-defined requirements for functionality, service levels, cost and delivery timing.
PP4	**Protection of Intellectual Property** The enterprise's intellectual property must be protected and reflected in the IT architecture, implementation and governance processes.
PP5	**On Demand** An enterprise's business processes must be integrated end-to-end with partners, suppliers and customers. A business must rapidly respond to any customer demand, market opportunity or external threat.
PP6	**Non-Functional Requirements Weighted Equally to Functional Requirements** Non-functional requirements will be designed, developed, tested and managed with the rigor of functional requirements.
PP7	**Flexibility** The IT architecture will incorporate flexibility to support changing business needs and to enable evolution of the architecture and the solutions built on it.
PP8	**General Governance** Compliance to the architecture and evolution of the architecture will be managed through controlled governance processes.
PP9	**Cost Performance** The IT architecture will be managed to ensure the cost effectiveness of the information and technology environment.
PP10	**Applications and Infrastructure Components** The application and infrastructure components will be designed and implemented in such a way as to facilitate monitoring and measurement.
PP11	**Service-Level Management** The IT architecture will support operation of business processes as defined by service-level agreements.

Table 8-4: Sample Principles for Process & Governance

Business Architecture Principles (BP):	
BP1	**Maximise Benefit to the Organisation** Information management decisions are made to provide maximum benefit to the enterprise as a whole.
BP2	**Business Continuity** Enterprise operations are maintained in spite of system interruptions.
BP3	**Common Use Applications** Development of applications used across the enterprise is preferred to the development of similar or duplicative applications provided only to a particular section/unit.
BP4	**Service Orientation** The architecture is based on a design of services for real-world business activities.
BP5	**Technology risk** Stability of business systems will be preserved through controlled usage and management of technology across its life cycle.
BP6	**Alignment of IT to business** The IT architecture will be aligned with the business vision, objectives, and strategies and will support the business operations.
BP7	**Strategic use of relationships** The IT architecture will leverage strategic relationships with other businesses and vendors to facilitate the building and evolution of the IT architecture.
BP8	**Optimize IT infrastructure** The IT infrastructure will be optimised based on business requirements and technology capabilities.
BP9	**Enforced Security Policy** Implement processes, procedures, and systems that promote enforcement of enterprise security policies.

Table 8-5: Sample Principles for Business Architecture

Information/Data Architecture Principles (IP):	
IP1	**Data are an Asset** Data are an asset that has value to the enterprise and is managed accordingly.
IP2	**Data are Shared** Data are shared across functions and units within the enterprise.
IP3	**Data are Accessible** Data are accessible for users to perform their functions.
IP4	**Data Trustee** Each data element has a trustee/owner accountable for data quality.
IP5	**Common Vocabulary and Data Definitions** Data are defined consistently throughout the enterprise, and the definitions are understandable and available to all users.
IP6	**Data Security** Data are protected from unauthorised disclosure and use.
IP7	**Application Independence** Applications are independent of specific technology choices and can operate on a variety of technology platforms.
IP8	**Ease-of-Use** Applications are easy to use with the underlying technology transparent to the users.
IP9	**Requirements-Based Access** A user (human or computer) should only be given enough privileges to do those tasks needed to perform a specified job activity, function, or task; no more, no less.
IP10	**Information Confidentiality** All components of the computing environment must maintain confidentiality and integrity of the information that is used to conduct business, with decisions based on data classification.

Table 8-6: Sample Principles for Information Architecture

Technology Architecture Principles (TP):	
TP1	**Requirements-Based Change** Changes to applications and technology are only made in response to business needs.
TP2	**Responsive Change Management** Changes to the enterprise information environment are implemented in a timely manner.
TP3	**Control Technical Diversity** Technological diversity is controlled to minimise the cost of maintaining expertise in and connectivity between multiple processing environments.
TP4	**Interoperability** Software and hardware conform to defined standards which promote interoperability for data, application and technology.
TP5	**Innovative and Agile** The IT architecture will readily support incorporation of new technologies to support business and technology innovation.
TP6	**Technology and Vendor Independence** The IT architecture will be designed to reduce the impact of technology changes on the business, as well as be resilient to change.
TP7	**Open Standards** The IT architecture will use open industry standards.
TP8	**Leverage Industry Knowledge** The IT architecture will leverage industry best practices.
TP9	**Testability** IT architecture should be designed for testing. Test environments will provide *simulation* of the production environment to a degree appropriate to the level and type of testing. The IT architecture should support test efforts that are able to work *independently*, without excessive coordination or scheduling.

Table 8-7: Sample Principles for Technology Architecture

Solution Architecture Principles (SP):	
SP1	**IT Responsibility** The IT department/unit is responsible for owning and implementing IT processes and infrastructure which enable solutions to meet user-defined requirements for functionality, service levels, cost and delivery timing.
SP2	**Single Point of View** Solutions provide a consistent, integrated view of the business, regardless of access point.
SP3	**Buy versus Build** Business applications, system components and infrastructure will be purchased unless there is a competitive reason to develop them internally.
SP4	**Integrated solutions** The IT architecture will support the delivery of business solutions composed of integrated application and infrastructure components.
SP5	**Completeness** A solution must be completely architected, including all viewpoints, solution portfolios and solution patterns.
SP6	**Reuse** Common components in the IT architecture should be used while balancing application and enterprise requirements. A solution reuses the as-built current state, unless there are capabilities that need to be updated.
SP7	**Real Solutions** The described solution should be delivered and running within the enterprise and is not just a piece of paper.
SP8	**Defence in Depth** Greater security will be obtained by layering defences. Security controls should be proportionate to risk.
SP9	**Security by Design** Security considerations should begin with the requirements phase of development and be treated as an integral part of the overall system design.
SP10	**Transparency** Security should be user transparent and not cause users undue extra effort.
SP11	**Limit Vulnerability** Design and operate IT systems to limit vulnerability and to be resilient in response.

Table 8-8: Sample Principles for Solution Architecture

8.3.4 Definition of Terms

A number of terms are used in the proposed REAM, including *inter alia* abstraction, framework, principle, metamodel, stakeholder, view and viewpoint. A description of all the relevant terms is provided in Addendum A.8-1.

8.3.5 Depictions

Various levels of the REAM were depicted in paragraph 8.2.3 – see Figure 8-6 for domains, Figure 8-7 for the current and future states, Figure 8-8 for different views, Figure 8-9 for relations and Figure 8-10 for the numbered areas. The actual size of each area is not significant as this is a topology and is dependent on the implementation by the specific enterprise. It is, however, beneficial to place more focus on the future state than the current state (Rosser, 2002: 1). The complete REAM depiction is seen below in Figure 8-11 (Note: RA indicates Reference Architecture). Three-dimensional images of the Relational EA Metamodel are displayed in Figure 8-12.

Figure 8-11: The Complete Relation Enterprise Architecture Metamodel

Figure 8-12: Photographs of paper models of the REAM (Roets, 2012)

8.3.6 Viewpoint Specifications

Viewpoints can be designed by using a basis template describing the means to construct an independent view. Viewpoints can be re-usable items stored in a library. An existing viewpoint can be customised for a specific architectural description (IEEE, 2000: 14). Each viewpoint specification relates to one specific view in an architectural description. Crystal (2007) states that viewpoints “are about applying a distinct point of view to help make sense of large information spaces”, but that they can also cause controversy and conflicts.

In the proposed REAM the views are depicted on the rectangular sides of the hexagonal prism – see Figure 8-8. The term view is used in the metamodel, but it indicates the view and its corresponding viewpoint specification. The viewpoint specifications will be addressed in areas B5-B16, I5-I16 and T5-T16. A viewpoint can, for example, be described in terms of the stakeholders, concerns, modelling technique (The Open Group, 2009c: 413) and the viewpoint language and analytic methods (IEEE, 2000: 17).

Some viewpoints are very close to being architectural domains, for example, Wegmann (2003) with the organisational level, the business level, the operation level and the technology level. Other examples of viewpoints are:

- Airport system: Pilot viewpoint *versus* air traffic controller viewpoint (The Open Group, 2009c: 416)
- Software development: the structural viewpoint *versus* the behavioural viewpoint (IEEE, 2000: 17)
- Zachman’s planner *versus* owner *versus* designer *versus* builder *versus* subcontractor *versus* functioning enterprise viewpoints (O'Rourke, Fishman & Selkow, 2003: 11)
- New small application: users’ viewpoint *versus* developers’ viewpoint (The Open Group, 2009c: 417).

8.3.7 Architectural Views

An architectural view provides a description of the entire system from a single perspective. The view is an instance of the template provided by the corresponding viewpoint specification. A view consists of attributes and models (IEEE, 2000: 15).

The views are flexible and not fixed or prescribed in the proposed REAM. The enterprise can implement as few or as many different views and label them as applicable to its environment. Therefore the REAM only refers generically to View A, View B, *etc*. Neaga & Harding (2005: 1093) recommend that the number be kept as low as possible. Holt (2009: 18) warns that views must have a purpose, be consistent and together provide a complete EA. A few possibilities are listed here:

- functional, physical and technical views (IEEE, 2000: 4);
- logical, process, physical, development views (Kruchten, 1995: 2);
- DoDAF's operational, systems and technical views (USA. Department of Defence, 2007a: 1-8);
- function, information, resource and organisation views (Rathwell, s.a.) and process view (Mingxin, 2009: 1425);
- technical, physical, logical views (Holt, 2009: 16); and
- Zachman's planner, owner, designer, builder, subcontractor and functioning enterprise views (O'Rourke, Fishman & Selkow, 2003: 11).

In the REAM views are depicted as hexagonal bands across all architectural domains along the sides of the hexagonal prism – for example, view A = B5, B6, B7, B8, T5, T6, T7, T8, I5, I6, I7 and I8. It is important to recognise that an architectural view includes inputs from all the architectural domains. Artefacts, practices and models from existing EA frameworks can be used to populate the viewpoint specifications.

8.3.8 Alignment Approach

The area of relating and aligning the different architectural domains was addressed in detail in Chapter 6. It resulted in a set of integration strategies and interlinking mechanisms (see Table 6-3). The integration strategies included two and three dimensional matrices, metamodels, viewpoints, service layers and reference and solution architecture. The interlinking mechanisms included human intervention, common language, iterative and re-usable processes, partitioning and blueprints.

The social aspects of alignment are also important. This entails management support for the strategy, initiatives and execution as well as staff support for management. Mechanisms such as a collective and collaborative planning style and open planning communications can be used to enhance the culture of working together. (Gregor, Hart & Martin, 2007: 111).

Taxonomy can be used as an alignment tool by providing a broader context and a hierarchy of concepts. In a taxonomy there is one fixed relationship, namely the parent or broader relationship in the hierarchy. In a thesaurus the relations are wider, albeit still closed, with for example, related terms and used or preferred term. Ontologies, however, have open vocabularies to describe the world in terms of a set of types, properties and relationship types. The language is defined by the creator of the description at will (Garshol, 2004: 384). According to Ohren (2004) an ontology consists of characteristic types with value sets as descriptors.

Hyam (2006) states that “Interoperability comes through semantics” and that the same semantics must persist across technologies. The ontology must be exposed in multiple ways (*e.g.* XSD (XML Schema Definition), GML (Generalised Markup Language), OWL (Ontology Web Language)), He is of the opinion that it is possible to construct an ontology based on a simple metamodel that can be mapped into multiple technologies, because

- an ontology is a collection of terms;
- each term has a denotation;
- a denotation is a description or a partial definition;
- each term has a connotation;
- a connotation is a definition; and
- a simple metamodel consists of:
    - o classes – the “is a” relationship,
    - o properties – attached values,

  - o relationships – properties with ranges of other classes,
  - o hierarchy – the ability to abstract, and
  - o simple instances – lists.

Gerber, Kotzé & Van der Merwe (2010: 63) came to the following conclusion: “it is clear that formal ontologies and the associated technologies can play a substantial role to enhance the quality of metamodels in enterprise architecture frameworks”. Further development of the REAM could include an ontology.

The Object Management Group (OMG) provides the Meta-Object Facility (MOF) as a common modelling languages for import and export and which can unify, for example, every step in the development of an application (Anon., 2012b). This is at a more detail level of abstraction than this discussion.

The REAM incorporates the following integration strategies:

- Four-dimensional matrices:
  - o Colours (horizontally) – architectural domains (BA, IA and TA)
  - o Horizontally – states and integration
  - o Vertically – views and viewpoints
  - o Colours (vertically) – level of detail.
- Metamodel
- Views & Viewpoints
- Reference architecture (B17-20, I17-20 and I17-20)
- Solution architecture (S1).

The REAM incorporates the following interlinking strategies:

- Partitioning – specific areas are dedicated to inter-relations (for example, B4 & T1).
- Common vocabulary – the artefacts in the integration areas (for example, I4 & B1) should be created jointly and result in a commonly understood terminology.
- Reusable processes (for example, viewpoints).
- Architecture processes (P1).

Examples of applying the chain links () in the REAM are tabled in Table 8-9.

Overview Level	
Link I4 & B1	Business information diagrams, describing the information needed to support a business service such as data consumed or produced by the business process and the source of the data
Link B4 & T1	Business footprint diagram, describing the links between business goals, units and services and mapping these on the technical components delivering the capability
Link T4 & I1	System/organisation matrix, depicting the relationship between application and organisational units
View Level *(for example, View A)*	
Link I8 & B5	Role catalogue, describing the authorisation levels or zones within the enterprise as well as the impact of system changes on roles for change management
Link B8 & T5	Locations diagram, providing a listing of all locations where the enterprise conducts business for testing proposed target solutions and for appropriate deployment strategies
Link T8 & I5	Processing diagram, describing deployable units of code/configuration and their deployment onto the technology platform in order to understand the components that form a deployable unit, how deployment units interconnect and how load and capacity requirements are generated for different technology components
Reference architecture level	
Link I20 & B17	System use case, describing the relationships between consumers and providers of application services
Link B20 & T17	IT/Business services matrix, describing what technology supports which business processes providing dependencies and drivers
Link T20 & I17	Application interaction matrix, describing the relationships among applications, and between applications and the physical technology components.
Note: The descriptions of the artefacts were taken from TOGAF (The Open Group, 2009c).	

Table 8-9: Examples of Application of Interlinking

8.3.9 Stakeholders and Concerns

The stakeholders (including the architect) and their interests and concerns should be documented (IEEE, 2000: 6). The stakeholders may differ depending on their role, the specific EA-project or the specific view. The stakeholder representation within the governance structures will therefore also differ. This should thus be addressed:

- in each area (B1-B20, I1-I20 and T1-T20) specifically;
- as part of the specific solution architecture (S1); and
- as part of the governance structures (P1).

The stakeholders should be managed in order to

- gain early input to shape the architecture;
- gain power to win more resources;
- create a good understanding of the architectural process and its benefits; and
- increase the effective anticipation of reactions (The Open Group, 2009c: 281).

The management of stakeholders entails identifying them, classifying them into roles/positions, determining the relevant management approach according to role/position, tailoring engagement deliverables and creating a stakeholder map (The Open Group, 2009c: 282).

8.3.10 Applying the Metamodel

In order for an organisation to be effective at enterprise architecture, it needs an enterprise architecture capability. This entails organisation “structures, roles, responsibilities, skills and processes” (The Open Group, 2009c: 16). The best way to apply the proposed Relational EA Metamodel is to follow an agreed process (part of area P1). The REAM is not intended to be prescriptive about the process to be followed, as each enterprise has its own priorities, challenges and process to adhere to.

“The TOGAF Architecture Development Method (ADM) provides a tested and repeatable process for developing architecture” (The Open Group, 2009c: 10). “TOGAF ADM is a comprehensive methodology that addresses architecture at the enterprise level as well as the individual system level. Activities in each phase of the ADM framework are well defined but it leaves implementation flexibility to practicing architects" (Tang, Han & Chen, 2004: 11). Steen *et al.* concur: "The TOGAF Architecture Development Method (ADM), developed by the Open Group provides a detailed and well-described phasing for developing an IT architecture" (2005). It is thus suggested that the ADM be utilised as a process to apply the proposed REAM.

The ADM enables the establishment of an architecture framework, to develop architecture content, to aid transition the organisation and to govern the realisation of the architectures. The ADM "process can be adapted to deal with a number of different usage scenarios, including different process styles" and also specific specialist architectures (The Open Group, 2009c: 213). This takes place within an iterative cycle of architecture definition and realisation for transition of the enterprise according to business goals. The ADM is divided into ten sections or phases (The Open Group, 2009c: 10), depicted in Figure 8-13 and is described in detail in the TOGAF Version 9 Manual (The Open Group, 2009c: 67). The ten sections or phases are listed with the relevant areas of the REAM in brackets:

- Preliminary Phase,
- Phase A: Architecture Vision (area P1),
- Phase B: Business Architecture (areas B1-B20),
- Phase C: Information Systems Architecture (areas I1-I20),
- Phase D: Technology Architecture (areas T1-T20),
- Phase E: Opportunities & Solutions (area S1),
- Phase F: Migration Planning (future state of BA, IA and TA),
- Phase G: Implementation Governance (area P1),
- Phase H: Architecture Change Management (area P1),
- Requirements Management (area P1).

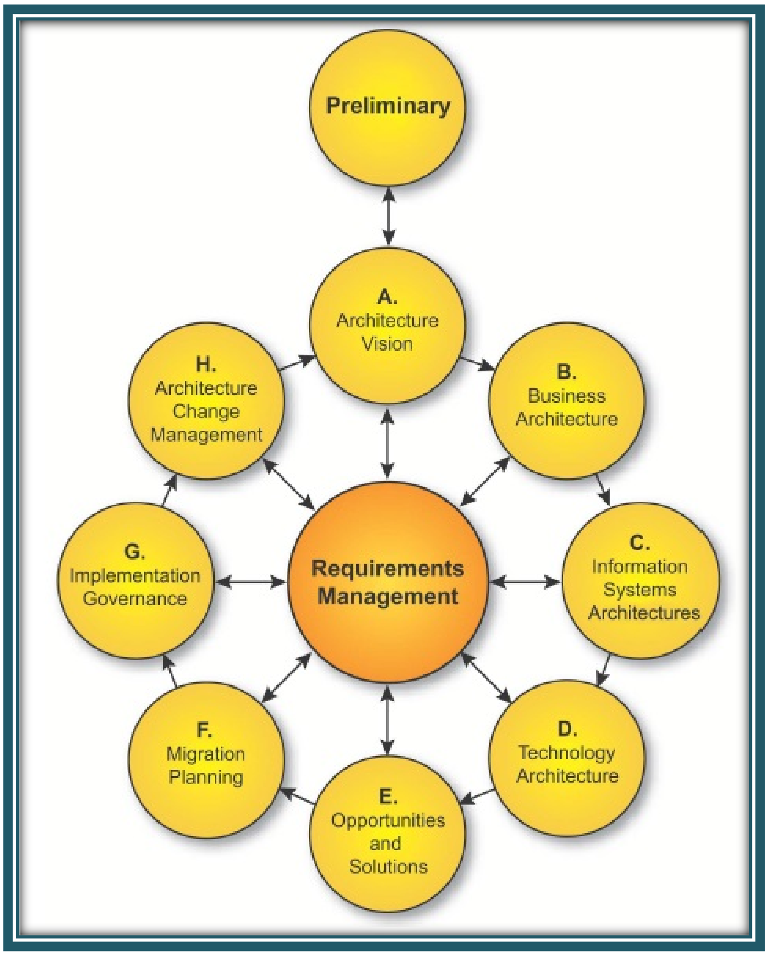


Figure 8-13: Summary of the phases of TOGAF's ADM (The Open Group, 2009c: 54)

Following the ADM comprehensively can be a huge undertaking. It is suggested that the following factors be taken into account to determine the optimal focus areas in order to gain the best value for effort for an enterprise:

- the strategic focuses of the enterprise,
- the business critical processes/applications,
- the areas with the highest investments,
- current or future projects,
- technology shifts and trends, and
- problem areas.

8.4 SUMMARY AND CONCLUSION

8.4.1 Summary

Five actions were taken to answer the sub-research question:

*j) How will an integrated metamodel, interlinking the different architectural domains, **be constructed** and described?*

First, the **relevant concepts**, from the explored existing frameworks, were extracted. The six significant ideas from existing frameworks, *i.e.* views, TOGAF processes, solution architecture, viewpoint classification, MOF processes and cube representation, as well as the other two useful artefacts, *i.e.* taxonomy and depiction, were taken into account to create and evaluate design alternatives.

Second,, three possible **design alternatives** were developed and evaluated:

- Design Alternative A – based on the previously proposed EA depiction (see Figure 8-2);
- Design Alternative B – based on Venn diagram principles (see Figure 8-3); and
- Design Alternative C – based on the CEiSAR cube principles (see Figure 8-4).

Third, aspects of the design alternatives were **combined** and **utilised** in order to benefit from the best properties of each design and to form the basis of the proposed Relational Enterprise Architecture Metamodel (REAM).

Fourth, the REAM was **developed**. The depiction of this metamodel is repeated here as a hexagonal prism net in Figure 8-14. The architectural domains business architecture, information architecture, technology architecture and solutions architecture as well as the architecture processes and governance are depicted in different colours. The different shades indicate the level of detail. The current and future states are depicted as vertical bands within each domain. Different views (including viewpoints) are depicted as horizontal bands across all the domains. Relations are indicated with chains between adjoining domains.

Figure 8-14: Proposed Relational EA Metamodel

Each area is numbered for easier reference. A summary of the areas is tabled in Table 8-10.

AREA	DESCRIPTION	AREA	DESCRIPTION
P1	Architecture processes and governance	I11	Future IA of View B
Business Architecture		I12	IA & BA part of View B
B1	Overview of BA and IA relation	I13	IA & TA part of View C
B2	Overview of BA current state	I14	Current IA of View C
B3	Overview of BA future state	I15	Future IA of View C
B4	Overview of BA and TA relation	I16	IA & BA part of View C
B5	BA & IA part of View A	I17	Detailed reference architecture of IA & TA relation
B6	Current BA of View A	I18	Detailed reference architecture of current IA
B7	Future BA of View A	I19	Detailed reference architecture of future IA
B8	BA & TA part of View A	I20	Detailed reference architecture of IA & BA relation
B9	BA & IA part of View B	**Technology Architecture**	
B10	Current BA of View B	T1	Overview of TA and BA relation
B11	Future BA of View B	T2	Overview of TA current state
B12	BA & TA part of View B	T3	Overview of TA future state
B13	BA & IA part of View C	T4	Overview of TA and IA relation
B14	Current BA of View C	T5	TA & BA part of View A
B15	Future BA of View C	T6	Current TA of View A
B16	BA & TA part of View C	T7	Future TA of View A
B17	Detailed reference architecture of BA & IA relation	T8	TA & IA part of View A
B18	Detailed reference architecture of current BA	T9	TA & BA part of View B
B19	Detailed reference architecture of future BA	T10	Current TA of View B
B20	Detailed reference architecture of BA & TA relation	T11	Future TA of View B
Information Architecture		T12	TA & IA part of View B
I1	Overview of IA and TA relation	T13	TA & BA part of View C
I2	Overview of IA current state	T14	Current TA of View C
I3	Overview of IA future state	T15	Future TA of View C
I4	Overview of IA and BA relation	T16	TA & IA part of View C
I5	IA & TA part of View A	T17	Detailed reference architecture of TA & BA relation
I6	Current IA of View A	T18	Detailed reference architecture of current TA
I7	Future IA of View A	T19	Detailed reference architecture of future TA
I8	IA & BA part of View A	T20	Detailed reference architecture of TA & IA relation
I9	IA & TA part of View B		
I10	Current IA of View B	S1	Solution Architecture

Table 8-10: Summary of Areas in the REAM

Lastly, the proposed REAM was **described** according to the headings below.

- Standards – the IEEE Std 1471-2000 was applied.
- Documentation – all architecture documentation should be structured and available in a repository. A template was provided (Table 8-1) and every area of the REAM was detailed.

- Architecture Principles – a template for describing architecture principles as well as a categorised list of example principles were provided.
- Definition of Terms – definitions for the relevant terms in the REAM were provided.
- Depictions – the metamodel was depicted as a geographical net and with photographs.
- Viewpoint Specifications – the how and what of viewpoints were discussed and examples were provided.
- Architectural Views – different examples and the application in the REAM were provided.
- Alignment Approach – the REAM's integration strategies and interlinking mechanisms were highlighted.
- Stakeholders and Concerns – stakeholders and their concerns form part of every area of the REAM.
- Applying the metamodel – the use of TOGAF's ADM is proposed as a mechanism for applying the REAM.

The sub-research question was thus answered by these five actions and concludes this iteration of the design and Section B.

8.4.2 Conclusion

The modelling process (Chapter 7) was followed to create three design alternatives, based *inter alia* on relevant input from existing EA frameworks. These alternatives and the experience gained were utilised to create the proposed Relational Enterprise Architecture Metamodel (REAM). The REAM was described using the following elements: standards, documentation, architecture principles, definition of terms, depictions, viewpoint specifications, views, alignment approach, stakeholders and concerns and mechanisms to apply the metamodel.

The REAM provides a multi-level, three-dimensional metamodel including different states, different views, reference architecture, solution architecture, inter-relations and architecture processes and governance.

This chapter concludes this iteration of the metamodel design as well as Section B. The next section, *Section C: Case Studies*, will entail the conceptual assessing of the Relational Enterprise Architecture Metamodel in one enterprise in each of three different vertical industries. The results of the assessment will be used as input to finalise the REAM.

SECTION C: EMPIRICAL RESEARCH – CASE STUDIES

The previous section contained the design of the proposed Relational Enterprise Architecture Metamodel for documenting an enterprise's enterprise architecture.

This section contains the empirical testing, evaluation and modification of the proposed REAM. The empirical studies section of the research will be documented in the following six chapters:

- Chapter 9 addresses the research design and methodology.
- Chapter 10 contains Case Study A conducted in an enterprise in vertical industry X.
- Chapter 11 contains Case Study B conducted in an enterprise in vertical industry Y.
- Chapter 12 contains Case Study C conducted in an enterprise in vertical industry Z.
- Chapter 13 contains the evaluation of the input received and a revised metamodel.
- Chapter 14 contains the applicability of the modified REAM to the University of Pretoria.

9 RESEARCH METHODOLOGY

9.1 INTRODUCTION

The objective of this section of the research is to assess the value and feasibility of the proposed REAM. The sub-research questions to be answered by the empirical research are:

k) *What is the **status quo** of enterprise architecture in the case study enterprises?*
l) *What are the **practical application possibilities** of the proposed integration metamodel within enterprises?*
m) *How was the proposed metamodel **received** in the case study enterprises?*
n) *What are the possible **limitations** of the proposed metamodel within the case study enterprises?*
o) *How was the proposed metamodel **modified**, based on the input from the case studies?*
p) *What is the applicability of the metamodel for the **University of Pretoria**?*

Before these questions can be answered, an appropriate research methodology has to be selected. The research methodology followed, will be described and motivated in this chapter. Hofstee (2006: 107) emphasises the importance of designing a research methodology. He provides criteria for deciding on a methodology, namely completeness, applicability, reliability, feasibility, ethicality, expense and time constraints. Both the methodology and design will be discussed in this chapter.

9.2 METHODOLOGY

According to Mouton's (2001: 143) classification of research designs, this research will fall into the broad category of empirical studies, because empirical studies are evidence-based with conclusions based on data that have been collected and analysed fairly (Yin, 2011: 21).

The study can be basic or applied research:

- basic research – aimed at developing, illustrating or testing general theories;
- applied research – aimed at providing knowledge to solve practical problems (Swanborn, 2010: 36). This study is applied research.

The research terminology and classifications used in literature differs, but this research methodology will be explained according to the classification by Edmonds & Kennedy (2013: 111) in Table 9-1.

<table>
<tr><td>Method</td><td colspan="30">Qualitative (versus quantitative or mixed)</td></tr>
<tr><td>Research</td><td colspan="30">Non-experimental (versus experimental and quasi-experimental)</td></tr>
<tr><td>Approach</td><td colspan="8">Grounded Theory</td><td colspan="7">Ethnographic</td><td colspan="8">Narrative</td><td colspan="7">Phenomenology</td></tr>
<tr><td rowspan="2">Design</td><td colspan="5">Systematic</td><td colspan="5">Emerging</td><td colspan="5">Constructivist</td><td colspan="5">Realist</td><td colspan="5">Critical</td><td colspan="5">Case Study</td></tr>
<tr><td colspan="6">Descriptive</td><td colspan="6">Explanatory</td><td colspan="6">Existential</td><td colspan="6">Transcendental</td><td colspan="6">Hermeneutic</td></tr>
</table>

Table 9-1: Qualitative Non-experimental Research Methods (adapted from Edmonds & Kennedy, 2013: 111)

9.2.1 Method: Qualitative

This study will be qualitative in contrast to quantitative or mixed method studies. The word *qualitative* implies a focus on quality of entities/processes/meanings which are not experimentally examined or measured in terms of quantity, intensity or frequency (Denzin & Lincoln, 2005: 1).

Denzin & Lincoln (2005: 3) describe qualitative research: “Qualitative research involves the studied use and collection of a variety of empirical materials – case study; personal experience; introspection; life story; interview; artifacts; cultural texts and productions; observational, historical, interactional, and visual texts – that describe routine and problematic moments and meanings in individuals’ lives”. Jupp (2006: 248) describes qualitative research as “Research that investigates aspects of social life which are not amenable to quantitative measurement” and “uses a range of methods to focus on the meanings and interpretation of social phenomena and social processes in the particular contexts in which they occur". This research will still be qualitative, although it will study business model phenomena rather than social phenomena.

The qualitative method of research was chosen, because it

- comprises a set of interpretive activities (Denzin & Lincoln, 2005: 6);
- focuses on understanding and on meaning (Edmonds & Kennedy, 2013: 112);
- provides critical interpretive and descriptive material (Denzin & Lincoln, 2005: 9);
- is used to explore the *how* and *why* of systems and human behaviour and what governs these behaviours (Edmonds & Kennedy, 2013: 112);
- provides a socially constructed image of reality (Denzin & Lincoln, 2005: 10);
- is used to describe phenomena in context making use of words instead of data (Edmonds & Kennedy, 2013: 112); and
- is inductive (Edmonds & Kennedy, 2013: 112).

9.2.2 Research: Non-experimental

This study will be non-experimental. "Non-experimental research is conducted when the researcher does not have direct control of the independent variables simply because their manifestations have already occurred" (Edmonds & Kennedy, 2013: 96). The variables can thus not be controlled through manipulation, inclusion, exclusion or group assignment. Statistical groupings can be utilised or causal relationships inferred from observational data. Kumar (2011: 113) contrasts non-experimental studies with experimental studies because they start from the effects to trace the cause and not the other way around.

Non-experimental research was chosen because:

- the purpose of the study is to describe a phenomenon (Maree, 2007: 34);
- no manipulation of the phenomenon is required (Maree, 2007: 152); and
- it explores the phenomenon (Maree, 2007: 152).

9.2.3 Approach: Phenomenological

This study will make use of the phenomenological approach. The phenomenological approach has the goal of understanding how individuals construct reality (Edmonds & Kennedy, 2013: 136) and how to understand social reality grounded in people's experiences (Gray, 2009: 22).

In contrast to applying an extensive research approach, in an intensive research approach "a researcher focuses on only *one* specific instance of the phenomenon to be studied, or only a handful of instances in order to study a phenomenon in depth" (Swanborn, 2010: 2). Each instance is usually called a *case*, where comparison takes place within the unit of observation. Mouton (2001: 143) classifies this as descriptive and the study within an enterprise as ethnographic.

The phenomenological approach was chosen, because:

- it "explores the meaning, composition and core of the lived experience of specific phenomena" (Edmonds & Kennedy, 2013: 136);
- it provides the subjective experience of the subject (Gray, 2009: 22);
- it "provides the framework for an in-depth analysis of a finite number of participants" (Edmonds & Kennedy, 2013: 138); and
- it seeks "the opinions and subjective accounts and interpretations of participants" (Gray, 2009: 28).

9.2.4 Design: Case Studies

This study will make use of case studies, with aspects of descriptive, explanatory, critical and constructivist design. The case study method is a scientific method of inquiry and creates an understanding of the *how* and the *why* by collecting in-depth good evidence through an intricate study from varied angles (Thomas, 2011: 4).

"Case studies are analyses of persons, events, decisions, periods, projects, policies, institutions or other systems which are studied holistically by one or more methods. The case that is the subject of the inquiry will be an instance of a class of phenomena that provides an analytical frame – an object – within which the study is conducted and which the case illuminates and explicates” (Thomas, 2011: 23). Yin (2009: 18) provides the following definition: “A case study is an empirical inquiry that

- investigates a contemporary phenomenon in depth and within its real-life context, especially when
- the boundaries between phenomenon and context are not clearly evident”.

The application of the case study method focuses on one thing (person, group, organisation, event, process, period, *etc.*) in detail and does not seek to generalise from it – it is about the particular. Sampling does not form part of the case study process, but the choice of the subjects of the case study should be justified. The choice could depend on proximity, involvement, being a good example or on being a different example. This can be viewed as a restricted sample to gain greater detail (Thomas, 2011: 3). Possible issues with case studies are the risk of losing focus, generalisation of results and subjectivity (Hofstee, 1984: 123).

The actors involved in a case may be located on the micro-, meso- or macro-level (Swanborn, 2010: 6). In this research the actors will be located on the meso-level or the organisational/institutional level.

A holistic multi-case design (Yin, 2009: 46) will be utilised. This refers to multiple cases but within different independent enterprises, *i.e.* not embedded in order to obtain compelling and complete results. This is, however, not sampling or an attempt to generalise, but a method to obtain replicated inputs from different angles. Each case will have an individual case report. After completion of these case reports, cross-case conclusions and modifications to the proposed model will be provided. According to Yin (2012: 131) a cross-case design addresses the same issues but more intensely.

Swanborn (2010: 150) provides the following steps in the case study process:

- gain permission for case study
- identify interviewee
- make an appointment in advance
- prepare structure and questions of interview

- travel to interview
- conduct interview
- organise relevant documents
- rework interview notes into a report
- supply report to interviewee
- discuss and finalise report
- analyse data.

The case study design was chosen, because

- it answers the *how* and *why* questions (Yin, 2009: 2);
- the investigator has little control over events (Yin, 2009: 2);
- it provides a rich (Thomas, 2011: 15) detailed (Hofstee, 1984: 123) picture;
- it focuses on a contemporary phenomenon within a real-life context (Yin, 2009: 2); and
- it fits the nature of the research question, containing descriptive and/or explanatory questions (Swanborn, 2010: 41).

Case studies will be discussed in more detail by addressing case selection, data gathering, analysis and interpretation and research ethics.

9.2.4.1 Case Selection

Multiple cases will be studied, but no sampling will be utilised in selecting the different cases. Case selection is first based on the different sectors, and then on obtaining a suitable and amenable company within each sector. The purpose of using different companies is to gain insight from different perspectives and it does not imply that the companies are representative.

Swanborn (2010: 45) provides the following steps for selecting cases:

- locate possible cases – list possibilities, take reputation and experts into account, use referrals from existing contacts or cases, post open applications via mass media;
- decide on number of cases – more cases are better, limited resources (time or money), determine value of each new case; and

- selection criteria – demarcation of domain, size of domain, pragmatic grounds, informative, representative, homogeneous or heterogeneous, developmental phase.

Swanborn (2010: 104) also provides questions to be raised when doing multiple cases:

- "Do we study and analyse these cases consecutively or simultaneously?" The determining factors are practical considerations, such as time, money, location and possible modifications to the approach.
- If consecutively, are the methods/techniques or questions altered according to experience? In order to negate this issue an initial pilot study could be used to give the finishing touches to the process. Consecutive or simultaneous cases can then follow on the pilot study. Preferably the results of the pilot study should not be included as a case study.
- Do we compare the results of the cases with each other? The criteria for selecting more than one case will influence the feasibility of comparing cases with each other, for example, the different cases are pure replication.
- "Do we aggregate the information from different cases?" Again, the selection criteria will indicate the aggregation of information, for example, cases chosen to represent different perspectives on the issue at hand.

Thomas (2011: 77) distinguishes between three kinds of cases based on their origin, namely key cases (good examples), outlier cases (interestingly differently) and local knowledge cases (personal experience). He also suggests a design mapping to indicate the different aspects of a case study. Figure 9-1 depicts an applied version of the diagram. The cases will be done as if in parallel, *i.e.* applying the same methodology and with the same set of interview questions. Due to the availability of respondents, time restrictions of the researcher and other practical factors the cases might be done sequentially in reality, but with a parallel approach.

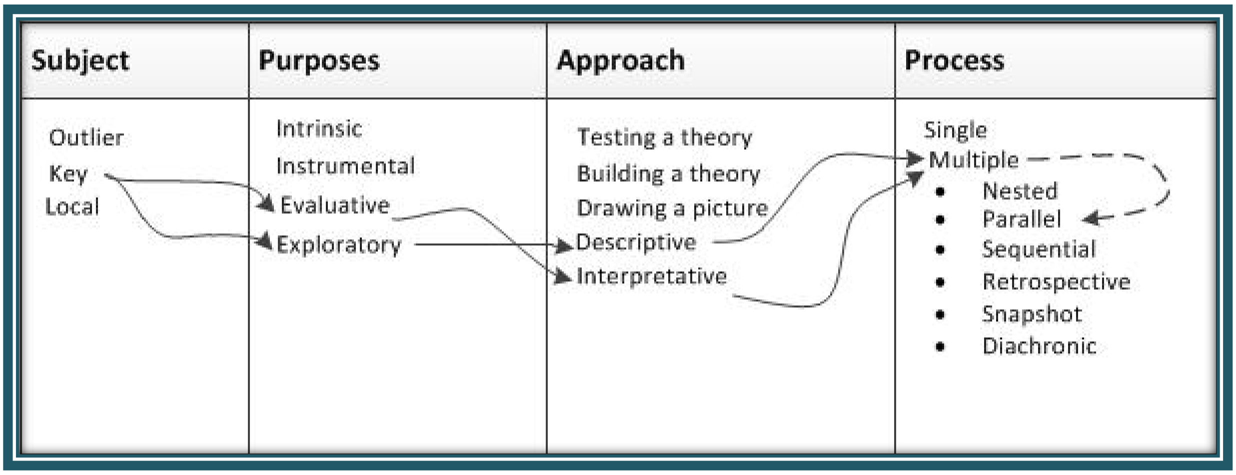


Figure 9-1: Adapted Research Design mapping (redrawn from: Thomas, 2011: 77)

9.2.4.2 <u>Data Gathering</u>

Case study data sources can include:

- documentation – efficient orientation method, *e.g.* agendas and minutes, reports, letters, proposals, archives;
- interviews – access to key personnel and then a broader group, as captured in, for example, interview transcripts;
- observation – complementary purpose and can be participatory or non-participatory, as captured in, for example, observer notes (Swanborn, 2010: 73); and
- physical sources – such as samples and materials (Mouton, 2001: 99).

Interviewing "is inextricably and unavoidable historically, politically, and contextually bound" and therefore never completely neutral (Denzin & Lincoln, 2005: 695). The interview is an active collaborative process in a context. Even the most carefully worded questions still hold a residue of ambiguity.

Interviews comprise:

- informants – key personnel or well-informed individuals approached via telephone, e-mail or contacts to gain information about the phenomenon, the processes and possible respondents; and
- respondents – relevant members of the enterprise to use in interviews to gain reliable descriptions of perceptions and input (Swanborn, 2010: 74).

Group interviewing "is essentially a qualitative data-gathering technique that relies on the systematic questioning of several individuals simultaneously in a formal or informal setting" (Denzin & Lincoln, 2005: 703). It has also been known as 'focus groups' especially in marketing research. Group interviewing can be used for testing a technique, identify key informants or triangulation and provides rich cumulative data, stimulates recall of participants and is flexible.

A semi-structured interview allows a list of issues on an interview schedule without a fixed order and allowing follow-up as necessary (Thomas, 2011: 163). According to Hofstee (1984: 135) an in-person interview can be structured (asking the same questions to everybody), including open-ended questions with individuals/groups.

Thomas (2011: 35) describes four kinds of questions, namely questions which

- describe the situation;
- clarify what is happening in a particular situation;
- determine what happens when a change is introduced; and
- examine the relation between things.

The set of questions should be piloted or pre-tested to refine the interview's effectiveness. Common mistakes to avoid are:

- ambiguous or vague items;
- double-barrelled questions;
- ignoring the effect of the order of the questions;
- questions outside scope of recipients' knowledge;
- leading questions;
- negative or double negative questions;
- too many questions; and
- sensitive or threatening questions (Mouton, 2001: 103).

Documenting the information from the different data sources could be done by taking notes, while participating, observing and listening and/or by recording and transcribing the conversations (Yin, 2011: 156). Triangulation within a case indicates the use of different data sources. This can, however, result in contradictory findings (Swanborn, 2010: 108).

Denzin & Lincoln (2005: 453) suggest documenting/recording the unusual as well as the ordinary. Yin (2011: 156) reminds the researcher to find a balance between documenting everything *versus* being too selective, to use drawings as an aid, and to obtain permission to record the conversations. He (Yin, 2011: 182) also promotes the use of a ‘database’ or an orderly set of records to keep track and organise all the notes and results.

9.2.4.3 Analysing and Interpreting Qualitative Data

According to Mouton (2001: 108) **analysis** “involves `breaking up’ the data into manageable themes, patterns, trends and relationships” in order to understand the relationships between concepts, constructs or variables. The **interpretation** involves the synthesis of the data into larger coherent wholes as well as relating the findings to existing models (Mouton, 2001: 109). Yin (2011: 176) provides the following steps: (1) compiling, (2) disassembling, (3) reassembling, (4) interpreting and (5) concluding.

Although Swanborn (2010: 115) distinguishes five traditional approaches to data analysis of case study data, listed below, he does not advocate following any one of these in full:

1 “Analysis of data collected in the field of changing organisations, according to Yin”– applied research with analysis by pattern-making, explanation-building, time-series analysis, logic modes and cross-case synthesis.

2 “Analysis of data collected in one of the qualitative traditions, especially grounded theory approach of Strauss and Corbin” – qualitative models for fieldwork.

3 “Data analysis and presentation according to the work of Miles and Huberman” – representational techniques such as tables, networks of geometrical figures and graphs.

4 “Time-series analysis” – statistical procedures with repeating measurements with a few precise variables.

5 "Data analysis according to Ragin's method" – Boolean logic, causal relations and fuzzy-set theory.

Swanborn (2010: 130) advocates reporting qualitative data in tables, with the following limitations:

- number of entries – combine the necessary number of entities to produce a clear and readable table;
- number of rows – maximum number between 15 and 20; and
- missing data – uniform, standard questions prevent missing and non-comparable data.

Tables should be followed by summaries per subgroup or per case (Swanborn, 2010: 132). Thomas (2011: 172) uses constant comparison, systems thinking, storyboards, developing theory and narrative as possible tools for analysing and interpreting the data.

9.2.4.4 Ethics

Research ethics is important, especially when working with people and enterprises. The nature of a case study – being closely involved with participants – makes ethics especially important. Ethics are the principles determining the right or wrong of conduct, but can be very complicated. These principles can include the right to take up people's time and energy, the possible discomfort of participants, the beneficiaries of the research, the privacy of the participants, the participants' standing, obtaining consent and confidentiality (Thomas, 2011: 68).

The impact of the research on the subject and the rights of the person or enterprise should not be infringed upon. Some areas of research ethics are:

- objectivity and integrity,
- recording of data,
- publishing ethics (plagiarism, ascription of authorship),
- accountability to society,
- responsibility to funders/sponsors of research, and
- right to privacy, anonymity and confidentiality (Mouton, 2001: 238).

Aspects to keep in mind in ethical research:

- Information participants need to know:
  - Nature and purpose of the study
  - Expected benefits
  - Possible harm
  - Confidentiality and anonymity
  - Ethics procedures
  - Researcher's detail
- Explanation of terms, *etc*.
- Option to take part (Thomas, 2011: 69).

9.3 DETAILED RESEARCH DESIGN

Based on all the above choices, this research consists of empirical, qualitative, non-experimental, phenomenological case studies.

Multiple cases will be researched. The **selection** of the cases does not represent sampling. The selection of possible enterprises is based on having a relative mature implementation of and expertise in enterprise architecture. The enterprises are in different sectors, to assist in obtaining as wide as possible a perspective. The enterprises are handled anonymously.

The case study **process** consists of the following steps:

- Approach potential enterprise:
  - *Purpose*: Obtain permission for collaboration or first contact session
  - *Participants*: Champion contact person in enterprise (informant) and researcher
  - *Method*: Electronic mail and telephonic conversation
  - *Input*: Introductory letter
  - *Content*: See Addendum A.9-1 for sample introductory letter
  - *Output*: Permission and an appointment for first contact session
- First contact session(s):
  - *Purpose*: Providing background to enterprise and gathering data on the enterprise's EA environment

- *Participants*: Informant(s) and researcher
- *Input*: Background on research, semi-structured interview questions and confirmation on planning of process
- *Content*: See Addendum A.9-2 for background summary and Addendum A.9-3 for semi-structured interview questions
- *Output*: Documentation from the enterprise and interview responses. For the sake of anonymity the male gender will be used throughout to indicate the informant and interviewees.
- *Note*: An iterative process might be needed, including electronic correspondence, to gather and understand the relevant data.

- Analysing and Applying the data gathered:
  - *Purpose*: Apply enterprise's EA to the proposed REAM (Chapter 8)
  - *Participants*: Researcher
  - *Input*: Data gathered during first contact session and follow-ups
  - *Content*: Summarised input and a table of REAM areas with applicable enterprise artefacts (for sample template see Addendum A.9-4). The questions and answers will be referred to by a code, consisting of the letter of the case study, the session number, a colon, a Q (question) or an A (answer) and the question number, for example, B1:Q3 will indicate question three of the first session with Case Study B. The summary could contain direct quotes from the interview/responses. The direct quotes will be italicised instead of using the normal apostrophes ("); the apostrophes will be used for complementary case study documents and will be cited by a code in brackets.
  - *Output*: An enterprise-specific application example of the REAM
- Second contact session:
  - *Purpose*: Feedback on applied REAM
  - *Participants*: A group of selected individuals from the enterprise, preferably including the informant (interviewees), and the researcher
  - *Input*: Introduction, the applied REAM and semi-structured interview questions
  - *Content*: See Addendum A.9-5 for session introduction and Addendum A.9-6 for semi-structured interview questions
  - *Output*: Interview responses
- Confirm summary of contact sessions:

  - *Purpose*: Obtain agreement on the summaries of the two contact sessions
  - *Participants*: Informant and researcher
  - *Input*: The summaries of the two contact sessions
  - *Content*: See Addendum A.9-7 for sample sign-off
  - *Output*: A signed sign-off per case study.
- Modify the REAM:
  - *Purpose*: Rework the REAM
  - *Participants*: Researcher
  - *Input*: Interview responses from second contact session
  - *Content*: Evaluation of proposed modifications by Case Studies
  - *Output*: A modified REAM diagram and area table
- Test applicability at the University of Pretoria in two sessions:
  - *Purpose:* Ascertain the possible applicability of the modified REAM at the University of Pretoria (UP)
  - *Participants:* Researcher, UP EA Team
  - *Input:* Background on the research, first session questions, UP applied REAM example, second session questions.
  - *Content:* Interview responses
  - *Output:* Recommendations

The possible limitations in this research are:

- limited number of cases;
- limited location of cases;
- limited comparison or generalisation between cases; and
- case/enterprise has too limited an understanding of EA to be able to evaluate the proposed REAM.

9.4 SUMMARY AND CONCLUSION

9.4.1 Summary

The objective of the research is to assess the value and feasibility of the proposed REAM. The sub-research questions to be answered by the empirical research are:

k) *What is the **status quo** of enterprise architecture in the case study enterprises?*

l) *What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

m) *How was the proposed metamodel **received** in the case study enterprises?*

n) *What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

o) *How was the proposed metamodel **modified**, based on the input from the case studies?*

p) *What is the applicability of the metamodel to the **University of Pretoria**?*

Before these questions can be answered an appropriate research methodology should be designed. After considering different methodologies, it was decided to utilise:

- the **qualitative** method of research, because it comprises a set of interpretive activities, focuses on understanding and meaning and provides critical interpretive and descriptive material;
- **non-experimental** research, because the purpose of the study is to describe and explore a phenomenon without manipulation;
- the **phenomenological** approach, because it explores the meaning, composition and core of the lived experience of specific phenomena and seeks the opinions and subjective accounts and interpretations of participants; and
- the **case study** design, because it answers the *how* and *why* questions, provides a rich and detailed picture and fits the nature of the research question, containing descriptive and/or explanatory questions.

The detailed research design was described by discussing the case selection, the first contact session, the data gathering and analyses, the application of the REAM, the second contact session, the sign-off and the possible limitations. The details are available in the addenda.

9.4.2 Conclusion

Based on the above methodological choices, this research consists of empirical, qualitative, non-experimental, phenomenological case studies done in three enterprises in different economic sectors, following the same methodology by means of semi-structured interviews. The researcher's conceptual application of the proposed REAM within the enterprise will be evaluated by the different cases/enterprises. The results will lead to modifications in the REAM, thus increasing its applicability and usability.

The Case Studies will be documented in the following chapters.

10 CASE STUDY A

10.1 INTRODUCTION

This first case study was conducted in Company A (who wishes to stay anonymous) in vertical industry Z – hereafter referred to as Case Study A. The sub-research questions to be answered by the different case studies are:

k) *What is the **status quo** of enterprise architecture in the case study enterprises?*
l) *What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*
m) *How was the proposed metamodel **received** in the case study enterprises?*
n) *What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The research methodology, as described in Chapter 9, was followed. The outcome of each step in the case study process for Case Study A will be described in this chapter.

10.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S)

Case Study A was identified in vertical industry Z as a company with relative mature enterprise architecture. Case Study A is a South African-based international company operating in 27 countries. It has ±49 000 employees and had headline earnings of R15,010 million in 2012. Its shares are divided ±40%/60% between non-public and public shareholders and ±54%/46% between South African and foreign shareholders (information gathered from Case Study A's website).

The correct contact person was identified through personal contacts via professional user group networking. An introductory letter (see Addendum A:9.1) was sent and an appointment for an hour and a half was set up as the first contact session.

10.3 FIRST CONTACT SESSION

A first contact session took place between this researcher and the contact person (informant) in Case Study A on 18 June 2013 for an hour and a half. The informant is the Group Enterprise Technology Architect for the enterprise. Confidentiality and anonymity were agreed upon. The informant signed the Informed Consent Form. Background on the research, based on Addendum A:9.2, was provided to the informant and a semi-structured interview based on the questions in Addendum A:9.3 was conducted. The following additional sample documentation was requested by the researcher and supplied by the informant to complement the interview: set of architecture principles (AD1), an example reference architecture (AD2), an example roadmap (AD3), the template for documenting a new solution/decommissioning of a solution (AD4) and a mandate for the IT governance committee (AD5). For the sake of anonymity, these documents will be cited by the code in brackets and not bibliographically.

The interview question responses, supplemented by documents, are summarised as follows:

A1:Q1 Please explain the enterprise architecture (EA) structure/set-up within your organisation with reference to business architecture, information architecture and technology architecture.

A1:A1 Four IT environments and three macro-architectures exist across the business units within the group. The structure has changed recently and the Group Enterprise Technology Architect is now formally responsible for all of these across the enterprise.

Each business unit has its own ExCo (Executive Committee) with a CEO and own investment committees, which feed into the overall group management. The **business** architecture and the **process information** architecture are done here by two business architects (on the ExCo), governed by the Enterprise Design Authority and the Program Enterprise Design Function. Their focus is on business processes. This domain is not as well established, with a maturity of about 2.5 out of 5.

The **application, information, data and infrastructure** architectures across all the business units are now the responsibility of the Group Enterprise Technology Architect.

He is also overall responsible for the IT strategy and IT architecture. There are about 100 full-time architects within his different teams. The application, information and data architecture domain is governed by Architecture Review Board(s) and Technical Review Board(s). The infrastructure/technology architecture is governed by an Infrastructure Group Standards Committee and an Infrastructure Group Design Authority. This domain is well established, with a maturity of about 3.2 out of 5. Concepts and conceptual logical models are the output provided to the design and build teams, which form part of the IT Shared Services under the direction of a CTO (Chief Technology Officer).

Interfacing with the less mature business architecture is problematic. In future more involvement with the design and build teams (part of IT shared services) is envisaged.
The CIOs (Chief Information Officers) are tasked with the Business/IT translation/alignment and reports to the COO (Chief Operating Officer) and have a dotted line to the CEOs.

A1:Q2 a) Do you make use of a recognised EA framework/model/methodology? b) If yes, how does this benefit your organisation, and c) how comprehensive is your implementation thereof?
A1:A2 A hybrid model is used, comprising TOGAF, Bredemeyer and home-grown practices – this is partly historic due to the previous federal organisation structure. The model is used as a framework for a way of thinking and as information for the skills base. *The EA framework provides a guideline for the organisation as far as how decisions are made, content required to make a decision and facilitate a common understanding of the governance within the organisation*. Less than 50% of TOGAF is implemented, but most the staff members are TOGAF certified.

A1:Q3 a) How do you govern the EA function within your organisation? b) For example, do you have a formal exception process and what does it entail?
A1:A3 Governance takes place on different levels:

- Technology (Infrastructure) Architecture – Infrastructure Group Standards Committee, Infrastructure Group Design Authority.

- Information (including data and application) Architecture – Architecture Review Board and Technical Review Board.
- Business and Process Information Architecture – Enterprise Design Authority and the Program Enterprise Design Function.
- Group management – a sub-committee of the IT Steering Committee is constituted as the IT Architecture Governance Committee and is responsible “to provide assurance that the current and envisaged IT architecture complies with the approved plan and that any deviations are appropriately escalated and resolved” (AD5).

Exceptions are handled by the Architecture Review Board according to a template presentation. The principles that business case prevails and buy *versus* build are followed. *The variety of historic federal ‘do it yourself’ applications bring complexity and integration challenges.* Tension exists between resource intensive core systems and the applications that business units need to perform. *Executive overwrites exist as well*.

A1:Q4 a) Do you have a set of architecture principles that has been agreed upon? b) If yes, how and for what purpose do you use them?
A1:A4 *A complete set of architecture principles does exist.* “Principles drive culture through behaviour so that consistent decisions are made”. “Principles help bridge the gap between what we do and how we do it” (AD1). Each principle is described in terms of: statement, rationale, implications, benefits, strategic alignment and consequences of violation.

The application thereof is not very strict (*laissez-faire approach*) and it is seldom referred to in the ARB (Architecture Review Board). The principles are revised annually. The principles *are used more as driving the culture and providing guidelines in practice for cross-checks and balances* and are not enforced. The principles are revisited once a year and must be approved by the CTO, CIOs, Group EA (IT Steering Committee). *There is a trade-off between the management culture and the enforcement of or adherence to principles*. These are not *enforced* but increases the transaction cost – *there is a price for democracy*.

A1:Q5 a) How is EA involved in your project management process? b) Do you make use of best practices/reference architectures or something similar?

A1:A5 *The first entry point of a project is an architecting phase as part of project initiation* (80% plus follow the correct process). EA has involvement through the different phases of a project:

- Planning – initially, EA is involved in the annual *guestimates*, planning and budgeting cycles.
- Business cases – architecture provides inputs.
- Initiation Phase – the initiation phase has a specific architecting phase – business roadmaps, technology roadmaps (5 year) and decommissioning of applications and systems.
- Stage gate/sign-off by EA and Project Review Board/Steering Committees.
- Design Phase – providing of solution architecture per release, trade-off between project scope and architecture with Project Review Board (the project managers also reports into the different CIO's). The design is measured against enterprise architecture.
- Build phase – sometimes referred back when issues arise.
- Final test phase – quick *dipstick test* to determine if architecture has been adhered to.

Reference architecture is done in some of the divisions, based on SOA (for about six years) and is rather stable. These best practices are *entrenched* in the organisation and are being referred to frequently. *Agility and flexibility are important*. For example, on a higher level an application architecture landscape has been depicted to indicate the technology choices and their status for all the different types of applications (AD1). Programme roadmaps are created containing: a four year plan, month-by-month timelines, colour-coded components, start and finish of actions, dependencies, *etc*. (AD3).

A1:Q6 a) Does EA form part of other organisational and/or decision-making processes? b) If yes, which processes are they?

A1:A6 Yes, EA forms part of:

- the annual planning and budgeting processes;

- the transformation of the business – *EA is used as a change agent* and to determine the impact of changed business models and processes, (*e.g.* cycle improvement program and the environment-provisioning program);
- the capacity planning process – EA influences the projections, impact and phasing out;
- operational monitoring process – *traditional monitoring is not sufficient anymore due to the layered architecture of systems;* a monitoring architecture is needed.
- HR (Human Resources) processes – complexity and multi-layered functioning of the EA structures change the skills mix required; especially in projects an integration manager is imperative; and
- the technology delivery management process – integration of elements.

A1:Q7 What mechanisms do you use to ensure consistent integration/interlinking between the different architectural domains?

A1:A7 The answer is threefold.

- First, the CIO's of the different business segments correlate business and IT architecture.
- Second, the Group Enterprise Architect is responsible to manage all the IT architecture across all the architecture teams and projects and report this to the business interface as the overhead architect.
- Third, the architecture team functions in a matrix. The matrix is a cross section of on the one hand business entities, initiatives and functions, such as transformation and integration; and on the other hand, EA, technology strategy, IT architecture and IT solution design. The enterprise architects handle the whole spectrum of technology strategy, IT architecture and IT solution design for a specific initiative, entity or project. Individuals within the matrix have a mixture of expertise focusing on architectural domains or segments, which are bigger than the specific initiative, for example, in IT architecture there can be *segment architects* focussing on security, integration, specific geographical area or a specific application. *Architecture modelling takes place on an event basis or on request*. The architectures are rolled up into the three macro-architectures to report to the Board. Conflict exists between the demand for pro-active architecture diagrams and the provisioning of resources to enable their creation.

A1:Q8 a) Is it necessary for you to distinguish between different views/viewpoints of EA to derive business value? b) If so, what are they?
A1:A8 *Absolutely*. It is definitely necessary and valuable to distinguish between different views. *There is a plethora of cuts necessary*. These cuts/views are based on specific focuses or events and addresses different domains depending on the individual's role within the organisation, such as a business audience, a very technical design audience or a planning project management audience. These are created by placing oneself in somebody else's shoes and add value to his situation. This is very time-consuming to draw up; therefore a ready-made set of artefacts does not exist and has to be created per requirement.

A1:Q9 How do you address, for instance, security in your EA framework/model?
A1:A9 Security is handled as a separate function in two ways, but is managed by the Group Information Security Officer, who operates on the same level as the different CIO's. There are business information security officers in a matrix configuration in each of the businesses.

First, IT security resorts under the Chief Security Architect. This combined entity addresses security as operational security, architecture and design, fraud and risk and security through all the disciplines. There is movement towards a federated approach, where *security will be built into the different disciplines and will not be managed as a bolted add-on*.

Second, the business has a risk function managing fraud and risk.

A1:Q10 How do you determine the architecture for the design and implementation of a new solution?
A1:A10 Solution architecting follows a process, starting with *high-level release domain scope architecture*. The *solution architecture* is then addressed on different levels with a dedicated specialist architect to coordinate and combine the different aspects of the solution into one solution architecture. This may include application architecture, security architecture, integration, *etc.* The solution is then reiteratively reviewed by the Technical Review Board and then approved by the Architecture Review Board. This

solution architecture is used for information purposes, adjustment in planning and setting of project scopes and determining estimates.

The next step is to derive technical solution specifications from the approved solution architecture. A technical solution specification is then drawn up per production line or product areas, as a factory model is being implemented for building the solution.

An example of an artefact describing the creation/decommissioning of a solution was provided, containing *inter alia*: a component model, a context model, as is patterns, processes, approach, principles, transition phase and the post migration diagram (AD5).

A1:Q11 a) Do you have architecture artefacts such as depiction(s) or document(s) which explain your organisation's EA? b) If yes, are you making use of any EA artefact repository/application/tool? c) If so, what are they?

A1:A11 A large collection of architecture artefacts exist. No life-cycle management tool is currently deployed. A list of different repository tools is being used throughout the process:

- Requirements: IBM® Rational® RequisitePro®[1] as a tool
- Process repository: ARIS (Architecture of Integrated Information Systems)[2]
- Technical architecture:
  - Standard Microsoft® SharePoint[3] system with *inter alia* Microsoft® Visio[4] (historically)
  - Atlassian[5] tools including Jira[6] for the repository as "open source tooling" for the technical artefacts
- Testing Tool : HP (Hewlett-Packard) Quality Center[7]

[1] For RequisitePro see http://www.ibm.com/software/products/us/en/reqpro/ [Accessed 24 July 2013].
[2] ARIS started as the academic research of Prof August-Wilhelm Scheer in the 1990s. It has an industrial background and has sold very well, becoming widespread. See http://www.softwareag.com/corporate/products/aris/default.asp [Accessed 24 July 2013].
[3] For Sharepoint see http://office.microsoft.com/en-us/sharepoint [Accessed 24 July 2013].
[4] For Vision see http://office.microsoft.com/en-us/visio/ [Accessed 24 July 2013].
[5] For Atlassian see https://www.atlassian.com/ [Accessed 24 July 2013].
[6] For Jira see https://www.atlassian.com/software/jira [Accessed 24 July 2013].
[7] For HP Quality Center see http://www8.hp.com/za/en/software-solutions/software.html?compURI=1172141 [Accessed 24 July 2013].

- CMDB (Configuration Management Database): multiple attempts have been launched to implement this properly, however, at the stage of the interview it is still in process
- A project runs a Microsoft® SharePoint infrastructure repository across the different architectural domains or tools, with reference to documents which for instance lies in the technology repository.

No global search engine or index is in existence – it is dependent on contact with the different domain experts to gain access to the relevant artefacts.

A1:Q12 a) Do you document both the current and the future state of architecture? b) If so, where does the focus/emphasis lie?

A1:A12 Both the current and the future state of the architecture are documented. The emphasis lies with the future state – approximately 80%. About 20% of the effort goes into documenting the current state. *The current state is documented to provide just enough understanding of the necessary context and to determine the magnitude and impact of changes.* This documentation is not used for business continuity or disaster recovery purposes – those documents are separate and form part of the production acceptance processes.

10.4 ANALYSIS AND APPLICATION OF GATHERED DATA

The gathered data were organised and summarised in Table 10-1 The left-hand columns summarise the data and the two right-hand columns reflect the application of the proposed REAM for Case Study A. This was used as input to the second contact session with Case Study A.

	Q#	Question Subject	Summarised Answer	OU/Process	Example artefacts
Dimension : Architectural Domains				P1: Processes & Governance	
P	Q1	EA set-up	Centrally managed in ETA team	Centrally managed ETA (Enterprise Technology Architecture) division	Organogram, job descriptions
	Q5a	EA in Project Management	Part of the different project phases	Project Management Process	Business case inputs, technology roadmaps, stage gates sign-offs, solution architectures, reference architectures
	Q5b	Reference Architecture	Reference architecture in some instances		
	Q6a	EA in other processes	Provides input to several organisational processes	Organisational processes: annual planning & budgeting, transformation, capacity planning, operational monitoring, HR, technology delivery management	Budget inputs, capacity planning inputs, *etc.*
	Q6b	List processes	Annual planning & budgeting, transformation, capacity planning, operational monitoring, HR, technology delivery management		
	Q3a	EA governance	Standards committee, design authority, ARB, technical review board IT Architecture Governance Committee	Governance bodies (standards commit-tees, design authorities, review boards) Architecture assurance to management	Architecture principles, Terms of Reference, Mandates
	Q3b	Exception handling	Exception process	Architecture exception process	Exception proposals (template)
	Q4a	Architecture principles	Set of annually reviewed principles	Culture of consistent decisions	Architecture principles
	Q4b	Purpose of principles	Guidelines in practices		
				B1-B20: Business Architecture	
BA	Q1	EA set-up	BA in business unit	Business Architecture in business units	Business process architecture
				I1-I20: Information Architecture	
IA	Q1	EA set-up	IA in ETA team	Information, data, application architecture in ETA division	Application/data standards and models
				T1-T20: Technology Architecture	
TA	Q1	EA set-up	TA in ETA team	Technology, infrastructure architecture in ETA division	Concepts, conceptual logical models
				S1: Solution Architecture	

SA	Q10	EA for solution	SA in ETA team	Multi-domain solution architecture, including decommissioning	Release domain scope, security architecture, integration architecture, technical solution specification
Dimension : Architectural States				Architectural States	
	Q12	Document current/future	Current – ad hoc, future – 80% of effort	Document current state as needed	Architecture diagrams with dependencies
				Document future state	Technology roadmaps
Dimension : Architectural Views				Architectural Views	
	Q8a	Distinguish views	Yes	Different versions according to audience as needed	Business audience brief, technical audience brief
	Q8b	List views	Event dependent and based on audience		
	Q9	Security	Matrix function, to be built into each function	Security view	Security in operations, architecture, design, fraud and risk
Dimension : Interlinking				Interlinking	
	Q7	Integration/interlinking	CIOs correlate BA & IA; Centralised Enterprise Architect coordinate all architecture efforts & liaises with business	Matrix teams, individual knowledge	Defined roles,
	Q5a	EA in Project Management	Input to planning, budget, business cases; architecting phase (project initiation); stage gates in project; final quick test	Project Management Process	Business case inputs, technology roadmaps, stage gates sign-offs, solution architectures, reference architectures
	Q6	EA in other processes	Input to and influence on organisational processes	Integrate EA into other organisational processes	Technology plan, technology delivery process, capacity projections
	Q11a	Artefacts	Yes		
	Q11b	Repository tool	Standalone array of tools		
	Q11c	List	RequisitePro, ARIS, Atlassian, SharePoint, Vision, Quality Center, CMDB	Repositories of artefacts	Standalone array of tools
Background					
	Q2a	EA framework/model	Hybrid: TOGAF, Bredemeyer & home-grown		
	Q2b	Benefit	Framework for thinking		
	Q2c	Comprehensiveness	Less than 50% of TOGAF		

Table 10-1: Data Analysis Case Study A

10.5 SECOND CONTACT SESSION

A second contact session was set up with the informant and three additional staff members on 12 December 2013 to evaluate the possible application of the REAM. The session was attended by the researcher, the informant and two additional staff members.

The session consisted of three parts.

First, the researcher delivered an introductory presentation which explained the REAM and its different dimensions (see Addendum A:9.5). During the presentation on the REAM, the following aspects of the metamodel were discussed/clarified:

- The area P1 represents the EA processes and governance and not the processes and governance of the whole organisation.
- There is vertical interlinking or traceability within the architectural domains through the different shades of colour, *i.e.* the different levels of detail.
- The researcher indicated that the numbering of the areas can be revisited, as it currently depends on the defined number of views. Views are flexible and will be not be completed by default.
- Multiple dimensions (five) are utilised, *i.e.* domain, level of detail, states, views and interlinking.

Second, the researcher delivered a presentation on the possible application of the REAM for Case Study A (see Addendum A:10.1). The application of the REAM is based on the analysis of the information gathered during the first contact session. During this presentation the panel indicated that some changes have taken place during the six months between the first and the second contact session. Additional information and clarification on their EA activities were provided. The essence of these modifications is:

- Processes & Governance – the design function was incorporated into the architecture section and a new governance model was communicated (Governance of Enterprise Architecture and Design (AD6)). This "is aimed at improving Governance of Technology Architecture and Design across" the company and embed "process-driven governance" to ensure "designed as architected" and "built as designed" solutions (AD6). The *blueprint design* and the technical design are now part of the ETA function. *A Technology delivery management function has been put in place to connect all the dots in the value chain with the blueprint as a basis*. The design process flows from architecture, to *solution architecture*, *technical integration design*, *technical solution design* (in the architecture section) *and then component design and built in the solution engineering team* (outside architect section). Human collaboration is unbelievably *challenging.*

- Architectural domains – Business architecture is not as mature as the other domains with outdated artefacts and too few people with relevant skills. *Information architecture is straggling* and does not exist *per se*, but is assigned to the business architects. Currently it is handled on a best-effort basis by the ETA division and an enterprise information management architecture is being created. Integration and interpretation is occupying a major part of the senior architects' time.
- Architectural states – *ninety percent of the effort goes into the creation of transitional states* or in other words the filling of the gaps between the present and the short, medium and long-term future. *An architectural forecasting model is expected by the organisation, but it is challenging to provide the blocks or initiatives to transition between the states.* The transitional states form part of the future state. The application of architecture has operational trade-offs, which need to be balanced out, for example, against project delivery.
- Architectural views – examples of views: *system component/context view, application landscape view, technical architecture view, sales (executive summary) view, deployment view, structural view, behavioural view, integration view and possibly a logging and monitoring view*. The architectural forecast for a longer period, as required by management, is challenging. There are a limited number of people that have the soft skills to translate, interpret, write, summarise or storyboard an *elevator pitch*. They have started to make use of iRise[8] to build *scenarios*, *visualisations, simulations and dependencies between loosely-coupled systems*. Security is now also part of ETA section as a cross-cutting view *to be built in instead of bolted on.*
- Interlinking – it is challenging to link and cross-reference artefacts and to keep them up to date. It is essential that people within the architecture teams, domains and business units talk to each other and collaborate continuously. A tool could be useful.

[8] For iRise see http://www.irise.com [Accessed 3 January 2014].

- Solution architecture – they make use of Bredemeyer Meta[9] and Bredemeyer Conceptual Architecture[10].

Third, after the two presentations, the questions, as provided in Addendum A:9.6, were discussed. The interview responses and discussions are summarised as follows:

A2:Q1 a) Please comment on the potential applicability of the REAM. b) What challenges do you foresee in the operational application of the REAM?

A2:A1 It is a complex model but a comprehensive abstract representation, which is more difficult to understand than the Zachman ontology. It will be a more practical model and will be easier to apply in a new or immature EA practice than the traditional models. It can be used very effectively internally to an EA organisation. *It is very good for use within the architecture community/practice, but will be difficult to use with stakeholders/executive to explain or request budget for an organisation's architecture function*. It can be used as an architecture checklist for every area to measure maturity/completeness of architecture. Customised tailoring takes place with all models/frameworks. The areas could be populated according to the organisation's needs and could assist in seeing *the wood from the forest*.

It is an operating model for the architecting community and is not to be used as an *architecture sales tool*. It is not and is not meant to be a methodology. It cannot be used for the whole architecture life cycle, but can be utilised as a type of scorecard. Every area should be described and have a checklist of possible artefacts. The purpose of the metamodel should also be defined.

A2:Q2 Please comment on the comprehensiveness of the REAM (covering all aspects of EA).

A2:A2 The REAM addresses every aspect of EA, for example, *planning*, *roadmaps*, *different levels of detail, business architecture*, *information architecture*, *application architecture, technology architecture, different states and views*. It could be interesting to cross-check the REAM with TOGAF to gain an understanding of the

[9] For Bredemeyer Meta Architecture see http://www.bredemeyer.com/pdf_files/ActionGuides/MetaArchitectureActionGuide.PDF [Accessed 3 January 2014].

[10] For Bredemeyer Conceptual Architecture see http://www.bredemeyer.com/pdf_files/ActionGuides/ConceptualArchitectureActionGuide.PDF [Accessed 4 January 2014].

comprehensiveness and to determine which sections of the life-cycle are not addressed. *The process and iterations thereof is not evident in the metamodel* as it does not include a methodology. The number and completeness of the different views are varying and not mandatory. It was suggested *that mandatory and/or optional areas should be indicated.* The metamodel covers the whole world, but the organisation can make choices relevant to their situation and context.

A2:Q3 Please comment on the ease of understanding and the ease of use of the REAM.

A2:A3 The metamodel *is complex and the devil is in the detail*, but the ease of use can be evaluated only after it has been applied. The determination of the correct level of detail will only be evident when applying the metamodel. A *noddy guide*, which indicates the start and a few steps and/or demonstrates the solution/architecture decision process based on a scenario, might be useful. The metamodel can be used as *a type of* archite*cture MIS* (Management Information System), which provides *snapshots to determine the status of an EA practice*. *The metamodel is perfect to plot different aspects of an organisation's EA such as organogram, work, initiatives, MIS artefacts, people, etc.* It can also be very useful as *a maturity evaluation for an auditor or consulting house*.

A2:Q4 a) The REAM provides explicit focus on interlinking. Does this provide value to the EA process? b) Does the application of the REAM improve the mutual influence between the architectural domains?

A2:A4 *Yes, interlinking is very important and should be addressed.* In contrast with traditional models, the REAM can lead to the discovery of new relationships and indicate linkages.

Interlinking is not, however, very obvious on the metamodel diagram – it lies in the detail of the pink links. The pink links indicate horizontal linking and the colour shading vertical linking. Traceability (or interlinking) could, for example, *also be required between the overview in a domain to the detail in another domain*. A suggestion is to build a 3-D (three-dimensional) REAM software model *with drill down into clickable areas, extracted mind maps and different layers of the metamodel*. Examples of a layer can be the interlinking aspects or people, artefacts, projects per area. The 3-D software model *can be used for visualisation – as construction architecture already does frequently*. This will be important for EA in general and specifically for the depiction of *multi-dimensional conceptual architecture models*.

A2:Q5 a) Could the metamodel contribute to improving the governance of EA? b) If yes, how could the metamodel facilitate the governance of EA?
A2:A5 *Yes, the REAM could contribute to EA governance.*

The metamodel could be used to evaluate architecture governance across all domains in terms of existence and compliance. It will not affect the governance process, but will assist in evaluating and monitoring the governance. *When the metamodel has been entrenched within an organisation, it will facilitate a common language*. The metamodel can contribute to the governance of sections of the architecture life cycle (ideation to deployment). *In the same way that a map of the world can portray different aspects (such as population, borders, produce, RAG* (Red, Amber, Green) *status*) *this metamodel can be used to portray (as a taxonomy) different aspects of EA* (*people, artefacts, initiatives, RAG status*). Another possible framework is the SABSA®[11] (Sherwood Applied Business Security Architecture) framework and methodology, which adds a security dimension to the Zachman Ontology.

A2:Q6 What modifications to the REAM would you propose to improve EA in your organisation?
A2:A6 The REAM provides a good overview. The following possible modifications/enhancements were discussed:

- provisioning of more detailed definition of the links;
- creating of a 3-D software model;
- indicating of excluded dimension(s), such as time (process and iteration);
- positioning and linking the REAM within the context of the organisation and with external influences (*e.g. organisational politics, finances, HR and technology, regulatory and economic trends*);
- proposing a methodology;
- investigate possibility of plotting a value chain as a view within the metamodel; and

[11] For SABSA® see http://www.sabsa.org [Accessed 8 Jan 2014].

- comparing the REAM to other industry-accepted models to deduce possible value.

10.6 MODIFICATIONS OF THE REAM

Case Study A provided possible modifications and enhancements to the REAM. The suggestions were categorised and are listed below:

- Depiction:
  - Enhance the visibility of the vertical interlinking or traceability between the different levels of detail within an architectural domain.
  - Revisit the numbering of the areas of the REAM, as the number of views will vary.
  - Indicate the mandatory and/or optional areas of the REAM.
  - Depict different levels of detail of the REAM.
- Clarification:
  - Clarify the inclusion of transitional states into the future-state dimension.
  - Clarify the purpose of the metamodel.
  - Clarify the dimensions not currently addressed (for example, time dimension, including iterations)
- Additional features:
  - Add a methodology or simple process guide.
  - Create a context diagram of the REAM within the organisation.
- Expansion possibilities:
  - Cross-check/compare the areas and dimensions of the REAM with TOGAF.
  - Develop a maturity evaluation method based on the REAM.
  - Build a 3-D REAM software model.

These proposed modifications and enhancements will be taken into account in modifying the REAM in Chapter 13.

10.7 SUMMARY AND CONCLUSION

10.7.1 Summary

The sub-research questions to be answered by, *inter alia,* Case Study A are:

k) *What is the **status quo** of enterprise architecture in the case study enterprises?*

l) *What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

m) *How was the proposed metamodel **received** in the case study enterprises?*

n) *What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The agreed-upon research methodology and case study process, as defined in Chapter 9, was followed with Case Study A in order to answer the above research questions:

- Identify enterprise and contact person – an applicable contact person and enterprise were identified and contacted.
- First contact session – a first contact session was conducted with the informant in the form of a semi-structured interview, based on the questions in Addendum A:9.3, to gain background and understanding of the EA function in Case Study A. The interview was complemented by relevant documentation from the enterprise. Case Study A has a well-established EA function, large architecture teams and centralised EA governance, with the exception of business architecture.
- Analysis and application – the EA operations of Case Study A was analysed (see Table 10-1) and the REAM was applied to it (see Figure A:10.1-1 through Figure A:10.1-6 in Addendum A:10.1).
- Second contact session – a second contact session was conducted with the informant and two other colleagues. First, the REAM was presented to the panel. Second, the REAM, as applied to their enterprise, was presented to the panel. Third, a semi-structured interview was conducted, based on the questions in Addendum A:9.6, to evaluate the REAM's applicability, comprehensiveness, ease of use, value of interlinking and governance with the aim of improving the REAM.
- Modified REAM – a list of possible modifications/enhancements to the REAM was gained from the process (see 10.6).

Case Study A views the **application possibilities** of the REAM to be a comprehensive abstract presentation of EA, which can be used very effectively within the architecture community or an architecture practice. It can be applied to plot different aspects of EA and measure the completeness, governance and maturity of EA in an enterprise. Case Study A views the **limitations** of the REAM diagram, as presented, to be not indicating transitional states explicitly, not covering the whole EA life-cycle, not providing a clear methodology and not providing sufficient detail regarding the pink links. Case Study A **received** the REAM enthusiastically and concurred that interlinking is a very important, and somewhat neglected, area of EA. The REAM sparked their interest to plot some of their EA aspects, such as governance. The suggested **modifications** include increasing the visibility of interlinking, mandatory/optional areas and transitional states, providing a context diagram and the addition of a simple methodology/process.

10.7.2 Conclusion

Based on the above outcomes of Case Study A, it can be concluded that Case Study A was conducted in a company with a relatively mature and established EA function. The REAM was well received and is viewed as a comprehensive, multi-dimensional EA metamodel, which can contribute to the interlinking between architectural domains and EA governance. It is flexible, but does not contain a methodology. A list of possible modifications/enhancements was gained from this case study.

Case Studies B and C will be documented in the following two chapters.

11 CASE STUDY B

11.1 INTRODUCTION

The next case study was conducted in Company B (who wishes to stay anonymous) in vertical industry Y – hereafter referred to as Case Study B. The sub-research questions to be answered by the different case studies are:

*k) What is the **status quo** of enterprise architecture in the case study enterprises?*

*l) What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

*m) How was the proposed metamodel **received** in the case study enterprises?*

*n) What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The research methodology, as described in Chapter 9, was followed. The outcome of each step in the case study process for Case Study B will be described in this chapter.

11.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S)

Case Study B was identified in vertical industry Y as a company with relatively mature enterprise architecture. Case Study B is a South-African company. It has ±54 700 employees and had headline earnings of R 4,186 million in 2013. It is wholly owned by the South-African government, but owns shares in a list of subsidiaries and joint ventures (information gathered from Case Study B's website).

The correct contact person – Chief Architect – was identified through personal contacts via a research organisation. An introductory letter (see Addendum A:9.1) was sent and an appointment was set up as the first contact session.

11.3 FIRST CONTACT SESSION

A first contact session took place between this researcher and the contact person (informant) in Case Study B on 30 April 2013 for an hour and a half. The informant is the Chief Architect for the enterprise. Confidentiality and anonymity were agreed upon. The informant signed the Informed Consent Form. Background on the research, based on Addendum A:9.2, was provided to the informant and a semi-structured interview based on the questions in Addendum A:9.3 was conducted. The following additional sample documentation was supplied by the informant to complement the interview: EA structure (BD1); Selecting of an architecture Framework for [Company B] (BD2); Architecture Maturity (BD5), Architecture Appeals Process (BD6), Generic SDLC for [Company B] ICT (BD7), System Development Life Cycle Process and RACI (BD8), [Company B] SOA Reference Architecture (BD9), Business case template and RACI (BD10), EA Deliverable Register (BD11), Information Architecture Services Catalogue (BD12), Entry Lower model (BD13), What is [Company B] EA (BD14). For the sake of anonymity, these documents will be cited by the code in brackets and not bibliographically.

The interview question responses, supplemented by documents, are summarised as follows:

B1:Q1 Please explain the enterprise architecture (EA) structure/set-up within your organisation with reference to business architecture, information architecture and technology architecture.

B1:A1 The architecture vision, as provided by the informant in an e-mail, dated 24 June 2013, is: “defining the overall form and function of information technology (business and IT) across [Company B] (including partners and organisations forming the extended enterprise), and providing frameworks, standards and guidelines for architectures”.

Architecture is one of six key management processes to ensure sustainable self-renewing performance (BD14). The following architectural domains have been defined:

- EBA – “Business strategies capabilities, processes and functions and EA governance”

- EIA – "Information, data, content and knowledge that supports [sic] the business processes"
- EAA – "Applications (whether purchased or built internally), services, components and utilities that automate business processes"
- ETA – "Technology infrastructure (whether insourced or outsourced)"
- Solution Architectures – "which incorporate and integrate many elements of these architectures in conjunction with projects" in "more detailed, implementation-specific models and designs"
- Enterprise Security Architecture – "integrates the components of all the above layers to satisfy security requirements" (BD1).

The architecture structure comprises twenty-seven persons, including the "Chief Architect", with:

- a "Principal Architect (Technical Architecture)", with a team of one "technology architect" and six "technology specialists";
- a "Principal Architect (Business Architecture)", with a team of six "business architects"; and
- a "Principal Architect (Information, Data and Security Architect)", with a team of three EA architects, one enterprise data, two information security, two integration, one enterprise content and one data quality architect (BD1).

B1:Q2 a) Do you make use of a recognised EA framework/model/methodology? b) If yes, how does this benefit your organisation, and c) how comprehensive is your implementation thereof?

B1:A2 **"**An eclectic approach to adopting an Enterprise Architecture Framework" (BD2) is in practice. The TOGAF framework was adopted and enhanced by the Gartner Enterprise Architecture Framework (GEAF). "No framework is an island and enterprise architecture is 'purpose-built' for a particular company", therefore it is recommended to adopt the framework with the best fit and augment it with other constructs and the organisation's architecture (BD2).

The following benefits of a framework are perceived:

- it is invaluable as a communication tool;

- it provides a "common definition of the content, scope and purpose of the architecture";
- it makes a "substantial contribution to communication and consensus building";
- it helps organising the complexity of architecture; and
- it "can encourage discipline in the development of the architecture and ensures a uniform approach by a team" (BD2).

TOGAF is implemented and the average maturity level is 2.24 with 5% of Case Study B at level 4-5 (BD3). According to an e-mail, dated 24 June, all "architects *are TOGAF 8.1* certified*"* and *"*Internal training provided on toolsets (ARIS) and methodologies*".*

B1:Q3 a) How do you govern the EA function within your organisation? b) For example, do you have a formal exception process and what does it entail?
B1:A3 *The EA is governed by an Architecture Review Board at the corporate level.* [Company B] *makes use of a federated governance model.* For example, a business unit *has its own review board called the Design Authority.* The "Design Authority is the steward of the standards framework and architecture principles and is responsible for resolving issues of non-compliance to the Enterprise Architecture in such a way as to keep the overall architecture evolving in a manner that best suits [Company B] over time*"* (BD5)*. In projects, the Design Authority issues a contract prescribing the technology to be used in a project in great detail*.

Exceptions are handled with the "architecture appeals process*"* (BD6)*.* This process is documented in detail and includes stakeholder involvement, the Design Authority, the ICT steering Committee and communication processes.

B1:Q4 a) Do you have a set of architecture principles that has been agreed upon? b) If yes, how and for what purpose do you use them?
B1:A4 *A set of architecture principles exists*, because "by documenting agreed Architecture Principles, and by having an Enterprise Architecture committee to enforce these principles, they are able to ensure that the ICT environment subscribes to a common charter that supports the goals of the business*"* (BD4)*.* Each principle is described by the following elements: domain, principle name, principle statement,

rationale and implications. The principles are further classified as “high level principles, business architecture, data/information architecture, application architecture, systems integration, technology architecture, security and telecommunication*”* (BD4)*.*

B1:Q5 a) How is EA involved in your project management process? b) Do you make use of best practices/reference architectures or something similar?
B1:A5 *Architecture is involved in all stages of the SDLC*. Case Study B makes use of a comprehensive ICT Generic SDLC Framework (BD7) and a SDLC process with RACI (Responsible, Accountable, Consulted, Informed) (BD8). This framework contains specific architectural outcomes for each of the 10 phases: “initiation, concept, planning, requirements analysis, design, development, test, implementation, operations and maintenance” and lastly “disposition” (BD7).

Case Study B makes *use of reference architecture,* for example “the SOA reference architecture” (BD8). *Compliance is tested against the reference architecture principles.*

B1:Q6 a) Does EA form part of other organisational and/or decision-making processes? b) If yes, which processes are they?
B1:A6 *Yes,* EA forms part of *technology investigations, strategic planning, budget, capacity-planning* and *portfolio management*. For example, the portfolio management process makes use of a “Business case template and RACI” (BD10), wherein the different architectural domains play a consultative role in all relevant sections of the business case. Case Study B makes use of an EA Deliverable Register (BD11), indicating all the deliverables outstanding for the different processes.

B1:Q7 What mechanisms do you use to ensure consistent integration/interlinking between the different architectural domains?
B1:A7 Anchor models (originating from the business) and a KPI (Key Performance Indicator) hierarchy are in place. The informant supplied the following mechanisms in an e-mail dated 6 Nov 2013:

- “Business Case – Architects (business, applications, data, security, technology and solution architects) meet weekly to discuss and assess solutions and alternatives recommended in the business case (cross reference). Joint signoff between architects with regard to the architectural section of a business case. Joint signoff between Portfolio Executive Manager, Production Manager, Chief architect and business owner of entire business case.

- Project Steering Committee Meetings – Architects are represented in these forums.
- The Design Authority – Issues architecture contract to the project manager, indicating standards, technology, *etc*., which must be used by the designers.
- SDLC: Requirements Analysis Phase Gate – Joint signoff between architects and business analysts. Business signs off the process and process prototype and/or the specification document.
- SDLC: Requirements Design Phase Gate – Joint signoff between the designers and the architects. Architects verify designs against approved standards, reference architecture principles, *etc*.
- Changes – Architects have to approve all changes at the change gates (co-signoffs)."

B1:Q8 a) Is it necessary for you to distinguish between different views/viewpoints of EA to derive business value? b) If so, what are they?
B1:A8 *Yes, it is.*

The informant provided an elaborate answer via an e-mail, dated 10 October 2013. The answer is summarised and quoted below. "Within each viewpoint, multiple levels of abstraction may exist, ranging from high level conceptual models, to detailed implementation level requirements, models and principles. At each of these levels of abstraction, multiple types of artefacts may be produced."

The following viewpoints are used:

- "Business Architecture Viewpoint – This represents business functions, processes and organisation
- Information Architecture Viewpoint – This represents elements such as information structure, information assets and information flow.
- Technology Architecture Viewpoint – This defines elements such as technical standards for software and hardware, middleware and infrastructure.
- Solution Architecture – This is the intersection of the other three viewpoints and is used to define how these viewpoints are applied to a specific solution. A solution is

the application services on the infrastructure that together automates the process and information required."

Different levels of abstraction are used within these viewpoints:

- "Conceptual – provides a consistent big picture view, used by senior management, focusing on specific concerns, but requiring some broader enterprise decision-making information. This should consist of short, high-level documentation and no products, vendors or implementation detail.
- Logical – provides an operationally focused enterprise architecture perspective to allow for better operational decisions. Operational management tends to need this perspective.
- Implementation – represents the granular details that enable change, whether for process improvement or system implementation. Change agents and project teams are predisposed to the implementation view. This should include not only vendor and product standards, but also the detailed guidelines needed to build the solution. In many cases, this level of detail will be generated in projects and incorporated with EA documentation as it is produced."

B1:Q9 How do you address, for instance, security in your EA framework/model?

B1:A9 Security is centralised at the head office to create security standards and policies and their enforcement across the whole organisation. "This often clash [sic] with projects undertaken by architects and application development teams in projects" (e-mail dated 26 Nov 2013). The scope comprises authentication, firewalls, DMZ, intrusion detection and prevention, antivirus software, encryption, patch management, security audits, availability, asset protection and security administration. Some of the operating divisions have their own security architect function which is aligned to the corporate one. These handle authorisation, assurance, risk management and typical security architecture artefacts. They make use of a security requirements checklist to verify that corporate security policies have been adhered to in all new projects and with change control on existing applications or infrastructure.

B1:Q10 How do you determine the architecture for the design and implementation of a new solution?

B1:A10 No answer was provided during the first session. Relevant information was provided during the second session, see 11.5.

B1:Q11 a) Do you have architecture artefacts such as depiction(s) or document(s) which explain your organisation's EA? b) If yes, are you making use of any EA artefact repository/application/tool? c) If so, what are they?

B1:A11 Yes, "We do not refer to EA directly with the business. The focus is getting business to work on their processes, and use the EA services that are provided to business through a measured SLA. The SLA addresses all aspects of the EA from business, information, data and security architectures" (e-mail dated 6 November 2013). An information architecture services catalogue (BD12) list all the available services and documents.

A number of tools are being used *to manage the architecture artefacts and support the business*:

- *ARIS*[12] on an Oracle 7.1 database – "business architecture, as-is documentation, future state, process models, functional models, information flow models, *etc*. ARIS services" are also "provided to the Lines of Business as a common business process modelling approach (currently contains more than 2000 models going back ten years)" (e-mail 7 November 2013).
- *Sybase PowerDesigner*[13] on a separate Oracle 7.1 database – "real time representation of all data and flow of data" (e-mail dated 7 November 2013).
- *TrouxView™*[14] on a separate Oracle 7.1 database – "linking all artefacts, initiatives and projects to strategy" (e-mail dated 7 November 2013).
- *Intranet* (SAP Portal) – "publish all standards, principles, patterns, and guidelines" (e-mail dated 6 November 2013).

[12] ARIS started as the academic research of Prof August-Wilhelm Scheer in the 1990s. It has as industrial background and has sold very well, becoming widespread, See http://www.softwareag.com/corporate/products/aris/default.asp [Accessed 24 July 2013].
[13] For Sybase PowerDesigner see http://www.sybase.com/products/modelingdevelopment/powerdesigner [16 November 2013].
[14] For TrouxView™ see http://www.troux.com/ [Accessed 16 November 2013].

The relevant users (role-based) are trained before gaining access to these tools/systems. *Quality assurance is done on all updates to models or processes before the changes are authorised* and published.

B1:Q12 a) Do you document both the current and the future state of architecture? b) If so, where does the focus/emphasis lie?

B1:A12 The informant responded in an e-mail, dated 26 Nov 2013: "[Company B] documents the future-state architecture only. After a few years the future-state architecture becomes the 'As-is' architecture and new future state 'To-Be' architecture is created". An anchor model or capability model is used to map the future-state capability map. A clearly defined and declared "Future Focused Business Architecture" is key to leverage reusable capabilities, assets and process execution.

11.4 ANALYSIS AND APPLICATION OF GATHERED DATA

The gathered data were organised and summarised in Table 11-1. The left-hand columns summarise the data and the two right-hand columns reflect the application of the proposed REAM for Case Study B. This was used as input to the second contact session with Case Study B.

	Q#	Question Subject	Summarised Answer	OU/Process	Example artefacts
Dimension : Architectural Domains				P1: Processes & Governance	
P	Q1	EA set-up	Central Chief Architect with 3 architecture teams	Central Chief Architect with three architecture teams	
	Q5a	EA in Project Management	Specific artefacts for every project phase	Project Management Process	Business case, architecture feasibility assessment
	Q5b	Reference Architecture	Reference architecture and compliance are tested	Compliance testing against reference architecture	SOA reference architecture
	Q6a	EA in other processes	Forms part of several organisational processes	Organisational processes: Technology investigations, strategic planning, budget, capacity planning, portfolio management	Business case, EA Delivery Register, planning artefacts
	Q6b	List processes	Technology investigations, strategic planning, budget, capacity planning, portfolio management		
	Q3a	EA governance	ARB, Design authorities	Governance bodies (ARB, Design Authorities)	Architecture principles, project contracts
	Q3b	Exception handling	Architecture appeals process through Design Authority (part of IT Steering Committee)		
	Q4a	Architecture principles	Set of principles enforced by EA Committee		
	Q4b	Purpose of principles	Ensure ICT subscribes to common charter	Ensure ICT subscribes to a common charter	Architecture principles
				B1-B20: Business Architecture	
BA	Q1	EA set-up	EBA in architecture team	Business architecture team within architecture structure	
				I1-I20: Information Architecture	
IA	Q1	EA set-up	EIA, EAA in architecture team	Information, data and security in team as part of architecture structure	
				T1-T20: Technology Architecture	
TA	Q1	EA set-up	ETA in architecture team	Technology & infrastructure in team as part of architecture structure	
				S1: Solution Architecture	
SA	Q10	EA for solution	No answer was provided	Unknown	Unknown

Dimension : Architectural States				Architectural States	
	Q12	Document current/future	Only future. Future becomes current state		
				Only future. Future becomes current state	Anchor models
Dimension : Architectural Views				Architectural Views	
	Q8a	Distinguish views	Multi levels of abstraction within different views – conceptual, logical & implementation	Conceptual, logical and implementation levels within business, information, technology and solution architecture	Business functions, information flows, technology standards & solutions
	Q8b	List views	Business architecture, information architecture, technology architecture, solution architecture		
	Q9	Security	ESA in architecture team	Centralised, with security architecture function in operational divisions	Security Requirements Checklist
Dimension : Interlinking				Interlinking	
	Q7	Integration/interlinking	Co-ordinated business cases, project steering, communication, design authorities, SDLC & change control	Joint sign-off, co-representation and input	Joint sign-offs
	Q5a	EA in Project Management	All phases of SDLC within SDLC framework with RACI	Project Management: SDLC process & framework with RACI	Specific architectural outcomes for each phase, RACI tables
	Q6	EA in other processes	Input into organisational processes	Integrate EA into other organisational processes	Technology investigations, strategic planning, budget, capacity planning, portfolio management
	Q11a	Artefacts	Yes		
	Q11b	Repository tool	List of tools		
	Q11c	List	ARIS, PowerDesigner, TrouxView, Intranet	Repositories of quality assured artefacts	Standalone / integrated
Background					
	Q2a	EA framework/model	TOGAF & GEAF (augmented)		
	Q2b	Benefit	Communication tool, common definition, organising the complexity, encourage discipline		
	Q2c	Comprehensiveness	Implemented at average maturity level of 2.24		

Table 11-1: Data Analysis Case Study B

11.5 SECOND CONTACT SESSION

A second contact session was set up with the informant and additional staff members on 25 February 2014 to evaluate the possible application of the REAM. The session was attended by the researcher, the informant and three additional staff members.

The session consisted of three parts.

First, the researcher delivered an introductory presentation which explained the REAM and its different dimensions (see Addendum A:9.5). During the presentation on the REAM, the following aspects of the metamodel were discussed/clarified:

- Solution architecture is the intersection of the other architectural domains.
- All EA artefacts should be managed by something similar to a CMDB.
- Transitional states are very important.
- Views should be based on stakeholders and context and can re-use the building blocks.
- The numbering of the REAM suggests a finite number of views – has to be reworked.
- Reference architecture provides detail and guidelines for solution architecture.
- Principles will be addressed in P1.
- Best practice will be addressed in the reference architecture, for example, B17 – B20.
- *It is a very interesting model* – it is difficult to implement a metamodel especially for a repository. Current repositories are only two-dimensional. Every area will have elements, information entities, activity, cross-reference, *etc*. This is a metamodel. It is not that easy to find the true relationships between the elements; and the toolsets have limitations in implementing the relationships.

Second, the researcher delivered a presentation on the possible application of the REAM for Case Study B (see Addendum A:11.1). The application of the REAM is based on the analysis of the information gathered during the first contact session. During this presentation the panel indicated that some changes had taken place since the first contact session. Additional information and clarification on their EA activities were provided. The essence of these modifications is:

- Processes and Governance – additional organisational process is business process management – the whole organisation is being re-engineered to focus on processes. All architectures are now supposed to be written from this perspective, based on best practices and automatic checks. Architecture is now playing a driving role within business. Process and Data Governance Boards have now been added
- Architectural domains – Business architecture has close relations with the strategy team, portfolio management team and the re-engineering team. Business is the owner of the artefacts, with EA being the custodian and fulfilling a strong consulting role. Focus moved from project-based to process-based organisation. Information architecture acts as glue or ties business and technology architecture together.
- Architectural states – Anchor models have now been supplemented by process modelling. Focus is applied to the short term to-be state.
- Architectural views – there are different interpretations regarding view and viewpoints The REAM is interpreted as stakeholder views. Security is centralised, based on COBIT (Control Objectives for Information and Related Technology) 5 and compliance testing is automated where possible. Compliance and controls are important.
- Interlinking – change/new business processes should in theory be verified by architecture before signed off and implemented. PowerDesigner is a modelling tool used for data, technology and application models with a live link to all systems, being updated in real time. This assists with impact analysis of changes. ARIS is used for business, process, logical and conceptual models. TrouxView is used as a repository of repositories.
- Solution Architecture – Case Study B *has not implemented solution architecture practices.* Development and business analyst teams tend to interfere with architecture solution roles. Business cases are drawn up jointly, which provides a few possible solutions as part of the portfolio base. Re-use of building blocks can be achieved by creating a joint single solution. It is problematic to obtain or train architects to be so cross-skilled across all the different architectural domains to be a solution architect. A joint group produces a better product.

Third, after the two presentations, the questions, as provided in Addendum A:9.6, were discussed. The interview responses and discussions are summarised as follows:

B2:Q1 a) Please comment on the potential applicability of the REAM. b) What challenges do you foresee in the operational application of the REAM?

B2:A1 The panel felt that more detail of the model was required to assess the applicability. It could be *very useful* if the detail is understood. *Detailed metamodel object view with object relationships will be important (and very useful) to make the model work* and to attain searchable objects. The model can be very useful if links are described. This seems to be an aggregation model, linking all the aspects. Different views of the model could be useful, for example, the business/logical view, a technical (implementation) view, domain role view and a user view. The links should not be explicit – for example, the logical view in business architecture doesn't correspond/link with the logical view of application architecture.

A 3-D model is probably needed. The current EA toolsets are not mature enough to service this model – the different domain architectures will be separately modelled by different tools.

B2:Q2 Please comment on the comprehensiveness of the REAM (covering all aspects of EA).

B2:A2 *It does* cover all the aspects and domains, but flexible numbering should make provisioning for *n* views.

Being a 3-D modelling, horizontal or vertical slices should be available, for example, how a strategy is linked with a solution and which solutions are linked to which strategy. A model should also assist in determining common infrastructure or re-usable building blocks. The maturity of the organisation will influence the acceptance of an EA model. Matching immature and mature areas within an organisation is challenging. The model can be extended to include the broader context, such as customers, partners, suppliers, enterprise to enterprise, *etc.* If an organisation is federal and the model is used in different units; the different instances of the model should be linked together. Any methodology can be used to populate the metamodel, for example, ADM or FEA methodology.

B2:Q3 Please comment on the ease of understanding and the ease of use of the REAM.
B2:A3 Different audiences will not understand the model in the same way, *architects can relate to this model and it makes immediate sense.* Providing guidelines, checklists and examples for each area of the model can assist with the operationalisation of the model – like the examples given in the Zachman ontology cells. The methodology used will impact on the ease of use.

B2:Q4 a) The REAM provides explicit focus on interlinking. Does this provide value to the EA process? b) Does the application of the REAM improve the mutual influence between the architectural domains?
B2:A4 The REAM has potential to improve interlinking, if the metamodel of objects and their relationships are described in detail.

The detail would lead to the discovery of linkages. It is difficult to determine the hand-over points or transitions between architectural domains. Translation between, for example, the business architecture language and application language or data dictionary, needs to take place. Outputs from activities in one domain could be used as inputs for an activity in another domain. The pink chains could describe these relationship points. At a higher level processes should be linked.

B2:Q5 a) Could the metamodel contribute to improving the governance of EA? b) If yes, how could the metamodel facilitate the governance of EA?
B2:A5 The metamodel could contribute to the governance of EA.

Inter-domain communication and expectations, as well as formal hand-over processes, will improve governance. Adding RACI role indicators will enhance governance.

B2:Q6 What modifications to the REAM would you propose to improve EA in your organisation?
B2:A6 The following modifications/enhancements were discussed:

- including various future transition states;
- clarifying the term ‘coherence’;

- indicating the dynamic size of the areas;
- providing examples in each area;
- providing more detail, such as object relationship view and detail of links;
- providing flexible numbering;
- indicating horizontal and vertical slicing;
- indicating the extended context (customers, partners, suppliers, enterprise to enterprise, multiple instances within an organisation, *etc.*); and
- adding RACI indicators.

11.6 MODIFICATIONS OF THE REAM

Case Study B provided possible modifications and enhancements to the REAM. The suggestions were categorised and are listed below:

- Depiction:
    - Provide flexible numbering of the areas.
    - Indicate dynamic size of the areas.
    - Include multiple transition states.
- Clarification:
    - Provide more detail, for example, an object relationship view and detail of links.
    - Clarify the term ‘coherence’.
- Additional features:
    - Indicate the extended context, for example, multiple instances within an organisation, customers, partners and suppliers.
    - Add RACI role indicators.
- Expansion possibilities:
    - Provide examples in each area.
    - Indicate horizontal and vertical slices.

These proposed modifications and enhancements will be taken into account in modifying the REAM in Chapter 13.

11.7 SUMMARY AND CONCLUSION

11.7.1 Summary

The sub-research questions to be answered by, *inter alia,* Case Study B are:

*k) What is the **status quo** of enterprise architecture in the case study enterprises?*

*l) What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

*m) How was the proposed metamodel **received** in the case study enterprises?*

*n) What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The agreed-upon research methodology and case study process, as defined in Chapter 9, was followed with Case Study B in order to answer the above research questions:

- Identify enterprise and contact person – an applicable contact person and enterprise were identified and contacted.
- First contact session – a first contact session was conducted with the informant in the form of a semi-structured interview, based on the question in Addendum A:9.3, to gain background and understanding of the EA function in Case Study B. The interview was complemented by relevant documentation from the enterprise. Case Study B has an established EA function with three teams and centralised EA governance.
- Analysis and application – the EA operations of Case Study B was analysed (see Table 11-1) and the REAM was applied to it (see Figure A:11.1- through Figure A:11.1-6 in Addendum A:11.1).
- Second contact session – a second contact session was conducted with the informant and three colleagues. First, the REAM was presented to the panel. Second, the REAM, as applied to their enterprise, was presented to the panel. Third, a semi-structured interview was conducted, based on the questions in Addendum A:9.6, to evaluate the REAM's applicability, comprehensiveness, ease of use, value of interlinking and governance with the aim of improving the REAM.
- Modified REAM – a list of possible modifications/enhancements to the REAM was gained from the process (see 11.6).

Case Study B views the application **possibilities** of the REAM to be dependent on the detail behind the high-level diagram. It could be used as an aggregation model and useful in depicting different views of an enterprise's EA. Case Study B views the **limitations** of the REAM diagram, as presented, to be not indicating transitional states, and having a lack of detail and methodology. Case Study B **received** the REAM as clear to architects and as input to discuss their own EA operations, for example, the hand-over or borders between architectural domains. The suggested **modifications** include provisioning of dynamic numbering and sizing of the areas, inclusion of the extended context and detailed examples and relationships.

11.7.2 Conclusion

Based on the above outcomes of Case Study B, it can be concluded that Case Study B was conducted in a company with a relatively mature and established EA function. The REAM was understood as a comprehensive metamodel, which can contribute to EA governance. It is on a high-level and does not have a prescribed methodology. Although Case Study B would prefer to comment on a more detailed version of the metamodel, the agreed upon, standard research methodology was followed. A list of possible modifications/enhancements was gained from this case study.

Case Study C will be documented in the following chapter.

12 CASE STUDY C

12.1 INTRODUCTION

This third case study was conducted in Company C (who wishes to stay anonymous) in vertical industry X – hereafter referred to as Case Study C. The sub-research questions to be answered by the different case studies are:

*k) What is the **status quo** of enterprise architecture in the case study enterprises?*

*l) What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

*m) How was the proposed metamodel **received** in the case study enterprises?*

*n) What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The research methodology, as described in Chapter 9, was followed. The outcome of each step in the case study process for Case Study C will be described in this chapter.

12.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S)

Case Study C was identified in vertical industry X as a company with relatively mature enterprise architecture. Case Study C is a South African-based company. It has ±21 000 employees and had consolidated operating revenue of R16,2 billion and profit after tax of R773 million in 2013. Its equity attributable to their owners was R21.5 billion as of 30 September 2013. Its shares are divided 50.3%/49.7% between government shareholders and public shareholders, of which 19.88% are from outside South Africa (information gathered from Case Study C's website).

The correct contact person was identified through personal contacts. An introductory letter (see Addendum A:9.1) was sent and an appointment for an hour and a half was set up as the first contact session.

12.3 FIRST CONTACT SESSION

A first contact session took place between this researcher and two contact persons (informants) in Case Study C on 6 March 2014 for an hour and a half. The informants are the Head of IT Strategy for the enterprise and the Head of Enterprise Architecture Solution Design. Confidentiality and anonymity were agreed upon. The informants signed the Informed Consent Form. Background on the research, based on Addendum A:9.2, was provided to the informants and a semi-structured interview based on the questions in Addendum A:9.3 was conducted. The following additional sample documentation was requested by the researcher and supplied by the informant to complement the interview: a Design Deliverable Document for [system] (CD1), Design Deliverable Document for [system] (CD2) and Managing the Systems Way: the Role and Impact of Technology (CD3). For the sake of anonymity, these documents will be cited by the code in brackets and not bibliographically.

The interview question responses, supplemented by documents, are summarised as follows:

C1:Q1 Please explain the enterprise architecture (EA) structure/set-up within your organisation with reference to business architecture, information architecture and technology architecture.

C1:A1 Information Technology has two departments, namely IT Solutions and IT Infrastructure. The Head IT Strategy is an executive of IT Solutions and also feeds into Group Strategy. The group strategy is a primary driver for IT strategy and alignment between IT and business is attained. IT is a critical enabler of company strategy. A CIO has recently been appointed for the first time.

EA Solutions Design, including IT systems/applications and company network, is a team of 36 architects. The focus is more on the Information Technology domain of EA. The enterprise architects are senior positions and link to the strategies as well as to the business architects. Previously there were domain architects and design architects in the EA teams, but now all these architects are solution architects. Every solution architect takes responsibility for a solution from the requirement or demand, through the whole life cycle to the solution design delivery. This is necessary to eliminate tunnel vision and silos. The whole process is very structured (CD1, CD2). Archimate®[15] is being used as language and modelling tool with UML (Object Management Group, 2014) as the standard for mapping business processes, for example, sequence diagrams. Case Study C has developed its own software for managing the life cycle through one methodology. The solution architects are mature IT people – new appointments are up-skilled and mentored. The benefits of having solution architects lies, *inter alia,* in the single point of contact for business and having one person that understands the whole solution end-to-end, its implications and obstacles. The added value of IT for the business is very important as IT is a large cost component of the business. Every solution should be able to prove its business value.

The business architects are a team of about eight people within IT, engaging with the strategy group and the business units. The plan is to move them into the business side.

C1:Q2 a) Do you make use of a recognised EA framework/model/methodology? b) If yes, how does this benefit your organisation, and c) how comprehensive is your implementation thereof?
C1:A2 TOGAF, including ADM, is being used in Case Study C. This is done within the ITIL and COBIT frameworks. They have expanded/customised the ADM with the help of a consultant company.

It is absolutely beneficial to make use of standard frameworks. Case Study C has a board decision to be King III[16] compliant. Frameworks are essential in achieving this and in providing checks and balances for audits. Architecture has to be structured as architecture decisions have financial implications.

TOGAF is being applied almost *militaristically* with processes through the innovation cycle. Almost all architects are TOGAF certified. TOGAF is being applied ±80%, with a few components left out due to overhead costs.

[15] For Archimate® see http://www.opengroup.org/subjectareas/enterprise/archimate [Accessed 8 March 2014].
[16] For King III Report see http://www.iodsa.co.za/?page=kingIII [Accessed 8 March 2014].

C1:Q3 a) How do you govern the EA function within your organisation? b) For example, do you have a formal exception process and what does it entail?
C1:A3 A demand management solution is in place to funnel and handle all the change requests and new demands/projects from the change control board. The alfabet[17] tool is being used to manage all the requests. Projects pertaining to change requests, capital projects as well as business as usual are handled. *The application of the methodology and documentation is very strict.* The whole life-cycle of a demand is managed and is auditable. Priorities are assigned to the projects, based on, *inter alia,* influence on revenue, market share and creation of new business as well as regulatory and legal requirements.

Every solution is governed (and measured) by the strategy and architecture principles and is based on the future mode of operation or the transitions in between. The first phase of EA governance takes place within the solution team by the team lead. Second, every solution has to be approved by the Global Design Authority. This body comprises business architect(s) and senior solution architect(s). There is a standard template to be used for documenting the solution. The design authority evaluates the solution from all angles, such as data architecture, migration architecture, security, system flow and user interface. After approval the solution needs to be procured, built or developed. This is a formal and minuted process. The minutes also serve as input to other bodies like the procurement committee. About 80% of business units follow this process but the rest is chaotic – this is being addressed by the new CIO. The convergence between network technologies and operational IT technologies complicates the process.

Exceptions will be approved if they are well motivated and linked to revenue generation. Exceptions can be expensive in the long term. There is now a focus on standardisation to assist the management of operational funds.

C1:Q4 a) Do you have a set of architecture principles that has been agreed upon? b) If yes, how and for what purpose do you use them?
C1:A4 *Yes, we have many architecture principles.*

[17] For alfabet see http://www.alfabet.com/en [Accessed 8 March 2014].

The principles are revised regularly to stay aligned with strategy. Principles are also used to measure if a solution is aligned with the decided strategic direction. Some examples are: best of breed, best of suite or organically grown solutions.

C1:Q5 a) How is EA involved in your project management process? b) Do you make use of best practices/reference architectures or something similar?

C1:A5 Case Study C has a large project management office, with diligent execution – projects are actually over-managed and the demand for EA artefacts is high. The life cycle management process contains architecture gates and quality documents, indicating the EA deliverables. Case Study C developed a SVC (solution value chain) (CD3), which is used to adhere to these gates, deliverables and quality requirements in each phase (analysis, design, build, pre-production, production, and acceptance). The project management office checks all the approvals and reviews as part of a formal process.

C1:Q6 a) Does EA form part of other organisational and/or decision-making processes? b) If yes, which processes are they?

C1:A6 *Yes*, EA form part of organisational processes.

EA is an integral part of the strategic planning process as well as the budgeting process for both capital and operational funds. The Head of EA is part of the top management of IT. Besides internal processes, EA plays a role and provides input into decision-making processes such as procurement and capacity planning.

C1:Q7 What mechanisms do you use to ensure consistent integration/interlinking between the different architectural domains?

C1:A7 There is a joint scoping of demands/projects/changes between the business architect management and the solution architect management. The DuPont Model[18] is used to determine ROI for the organisation. Co-signoff is required before the launch of a new project. The impact of a solution on the end-user is important due the size of the bargaining unit. Every solution impacting the staff has to be negotiated with the unions.

---

[18] For DuPont Model see http://en.wikipedia.org/wiki/DuPont_analysis [Accessed 8 Mar 14].

Until recently, IT was responsible for writing the business cases for business units to acquire capital for projects. The business is now the owner (and budget holder) of and responsible to write the business cases, with input from IT. This is reducing the number of unnecessary change requests received.

The solution architects are responsible to combine all the different architectural domains within the solution according to standards and best practises (reference architecture). Examples of tools/mechanisms used for standard integration between applications/systems are dipcode, web services, WebLogic[19] and even Tuxedo[20]. A waterfall approach with iterations is followed to produce the blueprint and solution design.

C1:Q8 a) Is it necessary for you to distinguish between different views/viewpoints of EA to derive business value? b) If so, what are they?
C1:A8 *Yes*, Case Study C distinguishes between different viewpoints.

All the Archimate views are developed, for example, stakeholder views (architecture stakeholders, business stakeholders and investment stakeholder), etc. All views are developed for each solution upfront after a high-level stakeholder analysis.

C1:Q9 How do you address, for instance, security in your EA framework/model?
C1:A9 Case Study C has an information security governance body outside of IT. They provide and assess the governance guidelines and policies. Within the EA team there are architects specialising in security. They consult to the solution architects on the creation of solutions. During the design process various security aspects are addressed, such as web portal, single sign on, application integration, role-based access, the user interface, business-to-business interface, firewall requirements and configuration. The information security governance body continuously assesses the security components of the solution. Security of information and the network is taken

[19] For WebLogic see http://www.oracle.com/us/products/middleware/cloud-app-foundation/weblogic/overview/index.html [Accessed 9 March 2014].
[20] For Tuxedo see http://www.oracle.com/us/products/middleware/cloud-app-foundation/tuxedo/tuxedo/overview/index.html [Accessed 9 March 2014].

very seriously and is addressed on C-level and downwards on various levels with respective policies. Every solution must adhere to all these policies and regulations.

C1:Q10 How do you determine the architecture for the design and implementation of a new solution?
C1:A10 The solution is determined by the solution architects according to a prescribed life-cycle, methodology and template. Refer to *C1:A1* for more detail.

The solution is documented and presented in a detailed standard document, containing for example, context, network architecture, as-is and to-be architecture, integration, assurance, migration, business architecture, solution alternatives, impact action patterns, activity diagrams, testing and quality of the solution (CD1). In another example the process patterns and sequence diagrams for each possible scenario were detailed (CD2).

C1:Q11 a) Do you have architecture artefacts such as depiction(s) or document(s) which explain your organisation's EA? b) If yes, are you making use of any EA artefact repository/application/tool? c) If so, what are they?
C1:A11 Several different types of artefacts exist.

A variety of tools are being used to create and/or save EA artefacts.

A taxonomy tool is used to record patterns in order to find reusable patterns to be used in certain conditions, such as MVC (Model View Controller) patterns in Java. Heuristics are used for unstructured metadata, especially regarding the history of what worked well. Artefacts are numbered and stored in the central corporate document management system, which has different authorisation levels and is being operated outside IT. Alfabet (mentioned in *C1:A3*) is used to store the primary building blocks. A single integrated search function of EA artefacts does not exist. A new project has been launched to provide document access via SharePoint[21] with, for example, a guide for new architects on how to find their way around and draw up a solution. Referring to the

[21] For SharePoint see http://office.microsoft.com/en-us/sharepoint [Accessed 24 July 2013].

strategy is important to determine the growth path or rationalisation path of a technology. Another useful tool is Enterprise Architect[22].

A master document with appendices is used to limit the length of each document. It assists the community of stakeholders in that they can easily access the relevant appendices instead of getting lost in the whole document.

C1:Q12 a) Do you document both the current and the future state of architecture? b) If so, where does the focus/emphasis lie?
C1:A12 *Yes, both states are documented.*

The solution requirements influence the documentation. In general, *equal effort* is put into the current and future state. A variety of documents are done on different levels. The current, future and transition states are documented. For example, strategy needs a current and future (five year) application landscape with the transitions in between.

12.4 ANALYSIS AND APPLICATION OF GATHERED DATA

The gathered data were organised and summarised in Table 12-1. The left-hand columns summarise the data and the two right-hand columns reflect the application of the proposed REAM for Case Study C. This was used as input to the second contact session with Case Study C.

[22] For Enterprise Architect see http://www.sparxsystems.com/products/ea/index.html [Accessed 8 March 2014].

	Q#	Question Subject	Summarised Answer	OU/Process	Example artefacts
Dimension : Architectural Domains				P1: Processes & Governance	
P	Q1	EA set-up	Managed in IT Solutions	Managed in IT Solutions via IT Strategy	
	Q5a	EA in Project Management	Integral part of life-cycle in projects. SVC	Solution Value Chain in project life-cycle	Deliverables according to phase
	Q5b	Reference Architecture	Yes, embedded in SVC	Reference architecture embedded in standards, templates, SVC	
	Q6a	EA in other processes	Yes	Organisational processes: strategic planning, budgeting, procurement, capacity planning	Minutes of approvals
	Q6b	List processes	Strategic planning, budgeting, procurement, capacity planning		
	Q3a	EA governance	Demand management, change control board, global design authority	Demand management, change control board, global design authority	Minutes, priorities
	Q3b	Exception handling	Motivation process via global design authority		
	Q4a	Architecture principles	Substantial set of architectural principles reviewed regularly		
	Q4b	Purpose of Principles	Measure alignment with strategic direction	Measure alignment with strategic direction	Architecture principles
				B1-B20: Business Architecture	
BA	Q1	EA set-up	BA in ITS, but not in ESA	Business architects in team within IT Solution, but separate from EA	
				I1-I20: Information Architecture	
IA	Q1	EA set-up	All architects are SA	Incorporated as solution architects	
				T1-T20: Technology Architecture	
TA	Q1	EA set-up	All architects are SA	Incorporated as solution architects	
				S1: Solution Architecture	
SA	Q10	EA for solution	All architects are SA	Solution architect is responsible for end-to-end solution	Solutions designs
Dimension : Architectural States				Architectural States	
	Q12	Document current/future	Equal effort, including transition states	Equal effort	Transition states

Dimension : Architectural Views				Architectural Views	
	Q8a	Distinguish views	Yes	Archimate views: architecture stakeholder, business stakeholder, investment stakeholder, business to business stakeholder, *etc.*	Documented up front
	Q8b	List views	Archimate views: architecture stakeholder, business stakeholder, investment stake-holder, B2B stakeholder, *etc.*		
	Q9	Security	Some SA specialises in security	Security specialists in Solution Design team	
Dimension : Interlinking				**Interlinking**	
	Q7	Integration/interlinking	Joint scoping & sign-off of projects between SA & BA. Inputs into business cases. Solution architects.	Joint scoping & sign-off of projects between SA & BA. Inputs into business cases. Solution architects.	
	Q5a	EA in PM	Project Management: SVC in life-cycle	Project Management: SVC in life-cycle	
	Q6	EA in other processes	Input into other organisational processes – major role in strategy	Input into other organisational processes – major role in strategy	
	Q11a	Artefacts	Yes		
	Q11b	Repository tool	List of tools		
	Q11c	List	Taxonomy tool, Heuristics, Alfabet, corporate document management system, SharePoint	Standalone array of tools to store artefacts	
Background					
	Q2a	EA framework/model	TOGAF & AADM within ITIL & COBIT		
	Q2b	Benefit	Compliancy, discipline, reusability		
	Q2c	Comprehensiveness	80% of TOGAF implemented		

Table 12-1: Data Analysis Case Study C

12.5 SECOND CONTACT SESSION

A second contact session was set up with the two informants and one additional staff member on 25 March 2014 to evaluate the possible application of the REAM. The session was attended by the researcher and the two informants.

The session consisted of three parts.

First, the researcher delivered an introductory presentation which explained the REAM and its different dimensions (see Addendum A:9.5). After the presentation on the REAM, they remarked that *the REAM was really very clever and they liked it (it can be put on a cereal box)*.

Second, the researcher delivered a presentation on the possible application of the REAM for Case Study C (see Addendum A:12.1). The application of the REAM was based on the analysis of the information gathered during the first contact session. During this presentation the panel provided additional information and clarification on their EA activities. The essence of these modifications is:

- Processes and Governance – The IT strategies are realised through enterprise architecture viewpoints/roadmaps, based on which solution architectures are designed, which are then realised through IT Solutions. *The relationship is thus: IT Strategy – Enterprise Architecture – IT Solutions.*
- Business Architecture – Business architects are part of the IT Solutions group.
- Architectural Views – the TOGAF stakeholder matrix is also used to identify stakeholders. Security is guided and monitored by the Corporate Information Security Governance Group.

Third, after the two presentations, the questions, as provided in Addendum A:9.6, were discussed. The interview responses and discussions are summarised as follows:

C2:Q1 a) Please comment on the potential applicability of the REAM. b) What challenges do you foresee in the operational application of the REAM?

C2:A1 *The mapping and numbering of areas are a good idea*. *The breakdown of the REAM can make a huge contribution towards the educating and up-skilling* of IT professionals for the architecture function – this is currently a challenge. The REAM is like a map to move through the architecture and determine gaps and quality. It can also be used to determine the maturity of each area. *The REAM has a lot of potential and the idea of the linkages is good.* The linkages/relationships can be enhanced by adding the security underpinning each link.

It was suggested that the views should be agreed upon upfront and fixed into the model. Another suggestion was to add other relations, such as 'affected' and 'affected by' to indicate the impact of areas on each other. For instance, if a business process is changed, what solution architectures and technical architectures should also change – it could even lead to the decommissioning of infrastructure/systems. Perhaps the metamodel can be put onto something like a soccer ball/stress ball as a visual reference.

C2:Q2 Please comment on the comprehensiveness of the REAM (covering all aspects of EA).
C2:A2 *The REAM is very complete*. Operational life cycle architectures, such as change management, release management, *etc.*, are not addressed, probably because a methodology is not provided or prescribed. It is important to be aware of the context in which architecture is applied. *It does not have to have a methodology; it can still be useful without it.*

C2:Q3 Please comment on the ease of understanding and the ease of use of the REAM.
C2:A3 The metamodel is very clear and fits on one page. It might be useful to build a physical or digital model. Although it is an enterprise architecture model, it provides a visual depiction to underline the importance of a business focus for every solution – an end-to-end solution. It depicts the necessity of addressing all the components, which is important for a total solution in contrast to a partial solution. Even though the detail might be technical and complex, it can be used to create better understanding of EA on other levels.

C2:Q4 a) The REAM provides explicit focus on interlinking. Does this provide value to the EA process? b) Does the application of the REAM improve the mutual influence between the architectural domains?
C2:A4 *Yes, definitely*.

Business architecture has not matured yet and is frequently housed separately. The model provides cohesion between the architectural domains. Enterprise architecture should focus on solutions for the business and IT should position the solution – *it is all about business.* A system also makes use of interlinking, as the components are dependent on each other. *Like anything in life, EA should have inflows and outflows to prevent becoming stagnant*.

C2:Q5 a) Could the metamodel contribute to improving the governance of EA? b) If yes, how could the metamodel facilitate the governance of EA?
C2:A5 They agreed that the model can assist with the governance of EA. The model can be used to do a pre-evaluation of an artefact to determine if it is complete. The area numbers can be used by members of the governance boards which are not directly *au fait* with enterprise architecture. In this way a fool-proof mechanism for governance can be applied.

C2:Q6 What modifications to the REAM would you propose to improve EA in your organisation?
C2:A6 *The REAM provides a good overview and is a clever, innovative and refreshing model.* The following possible modifications/enhancements were discussed:

- adding the ‘affected’ and ‘affected by’ relationships;
- being more specific with the views/viewpoints;
- indicating transitional states;
- adding a security component to the linkages;
- providing a physical model; and
- providing context to the model.

12.6 MODIFICATIONS OF THE REAM

Case Study C provided possible modifications and enhancements to the REAM. The suggestions were categorised and are listed below:

- Depiction:
  - Indicate the transitional states.
- Clarification:
  - Provide specific views/viewpoints.
- Additional features:
  - Add 'affected' and 'affected by' relationships.
  - Add a security dimension to the linkages.
  - Provide context to the REAM.
- Expansion possibilities:
  - Provide a physical model of the REAM.

These proposed modifications and enhancements will be taken into account in modifying the REAM in Chapter 13.

12.7 SUMMARY AND CONCLUSION

12.7.1 Summary

The sub-research questions to be answered by, *inter alia,* Case Study C are:

k) *What is the **status quo** of enterprise architecture in the case study enterprises?*

l) *What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

m) *How was the proposed metamodel **received** in the case study enterprises?*

n) *What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The agreed-upon research methodology and case study process, as defined in Chapter 9, was followed with Case Study C in order to answer the above research questions:

- Identify enterprise and contact person – an applicable contact person and enterprise were identified and contacted.
- First contact session – a first contact session was conducted with the two informants in the form of a semi-structured interview, based on the questions in Addendum A:9.3, to gain background and understanding of the EA function in Case Study C. The interview was complemented by relevant documentation from the enterprise. Case Study C has a well-established EA function, large architecture teams and centralised EA governance.
- Analysis and application – the EA operations of Case Study C was analysed (see Table 12-1) and the REAM was applied to it (see Figure A:12.1-1 through Figure A:12.1-6 in Addendum A:12.1).
- Second contact session – a second contact session was conducted with the informant and two other colleagues. First, the REAM was presented to the panel. Second, the REAM, as applied to their enterprise, was presented to the panel. Third, a semi-structured interview was conducted, based on the questions in Addendum A:9.6, to evaluate the REAM's applicability, comprehensiveness, ease of use, value of interlinking and governance with the aim of improving the REAM.
- Modified REAM – a list of possible modifications/enhancements to the REAM was gained from the process (see paragraph 12.6).

Case Study C views the **application possibilities** of the REAM to be enhancing the understanding and skill level of EA, plotting EA quality, gaps and maturity and forming strong linkages between the architectural domains. Case Study C views the **limitations** of the REAM diagram, as presented, to be the lack of providing specific viewpoints as well as operational context. Case Study C **received** the REAM enthusiastically and called it clever, innovative and refreshing. The suggested **modifications** include strengthening the linkages to indicate security and the 'affected' and 'affected by' relations, providing a physical model and indicating context.

12.7.2 Conclusion

Based on the above outcomes of Case Study C, it can be concluded that Case Study C was conducted in a company with a relatively mature and established EA function. The REAM was understood as a comprehensive metamodel, which can contribute to EA governance, maturity mapping and education. Emphasis was placed on the REAM's ability to enable full end-to-end solutions, addressing all the relevant components. A list of possible modifications/enhancements was gained from this case study.

The modification suggestions from Case Studies A, B and C will be used for enhancing the REAM in the next chapter (Chapter 13).

13 EVALUATION AND REVISED METAMODEL

13.1 INTRODUCTION

The Case Studies (in the previous three chapters) evaluated the REAM and provided input to enhancing and/or modifying the REAM. These inputs will now be evaluated and used to modify the REAM. This will provide the answer to the next sub-research question:

*o) How was the proposed metamodel **modified**, based on the input from the case studies?*

13.2 CASE STUDY INPUTS

The inputs from the different Case Studies were categorised and listed. A combined list of enhancement suggestions was compiled. The number of instances of a specific enhancement suggestion is indicated in brackets.

- Depiction:
  - Indicate multiple transitional states in the future state. (3)
  - Revisit the numbering of the areas of the REAM, as the number of views will vary and should be flexible. (2)
  - Enhance the visibility of the vertical interlinking or traceability between the different levels of detail within an architectural domain OR indicate horizontal and vertical slices. (2)
  - Provide/depict more detail or different levels of detail, for example, an object relationship view and detail of links. (2)
  - Indicate the dynamic size of the areas. (1)
- Clarification:
  - Clarify the term ‘coherence’. (1)
  - Clarify the purpose of the metamodel. (1)
  - Clarify the dimensions not currently addressed (for example, time dimension, including iterations). (1)
- Additional features:

  - Provide context to the REAM, for example, a context diagram within the enterprise or even indicate the extended context, for example, multiple instances within an enterprise, customers, partners and suppliers. (3)
  - Add a methodology or simple process guide and indicate the mandatory and/or optional areas of the REAM. (2)
  - Add ‘affected’ and ‘affected by’ relationships to linkages. (1)
  - Add a security dimension to the linkages. (1)
  - Add RACI role indicators. (1)
- Expansion possibilities:
  - Provide a physical model or build a 3-D software model of the REAM. (2)
  - Provide specific views/viewpoints. (1)
  - Provide examples in each area. (1)
  - Cross-check/compare the areas and dimensions of the REAM with TOGAF. (1)
  - Develop a maturity evaluation method based on the REAM. (1)

13.3 EVALUATION OF INPUTS AND MODIFICATIONS OF THE REAM

Each of the above-mentioned inputs will be addressed individually.

13.3.1 <u>Depiction Inputs</u>

The following depiction inputs were received, evaluated and addressed.

13.3.1.1 <u>Transitional States</u>

All three Case Studies indicated that they make use of multiple transitional states. This depends on the interpretation of the term ‘future state’. The future state can be seen only as the ultimate goal or it can include multiple transitional states such as the short, medium and long-term future.

This is an important input and will be addressed by changing the diagram to indicate the short, medium and long-term future – see Figure 13-1 for these changes and Figure 13-

7.for the composite modified REAM. However, more transitional states can be added when required.

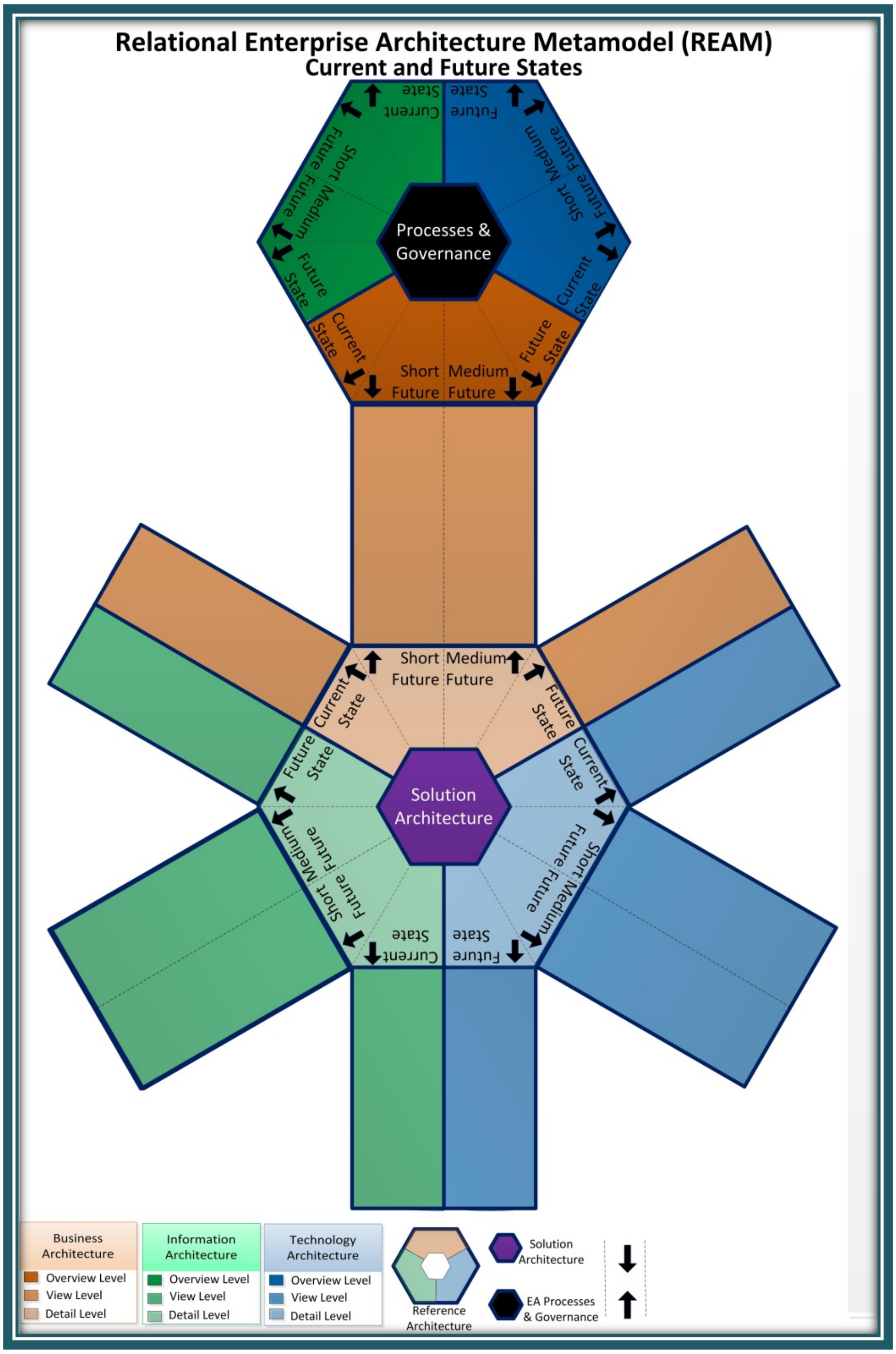


Figure 13-1: REAM with Changes to States

13.3.1.2 <u>Area Numbering</u>

In two of the Case Studies (A and B) the numbering of the areas was discussed. Currently, the numbering is based on three views only. This is in conflict with the REAM concept of being flexible and handling a varying number of views depending on the enterprise, situation or solution.

This is a valid input and the numbering of the areas on the diagram will be changed by using small letters to indicate the number of views and by moving the open-ended numbers to the middle area of the diagram – see Figure 13-2 and Figure 13-3 for these changes and Figure 13-7 for the composite modified REAM.

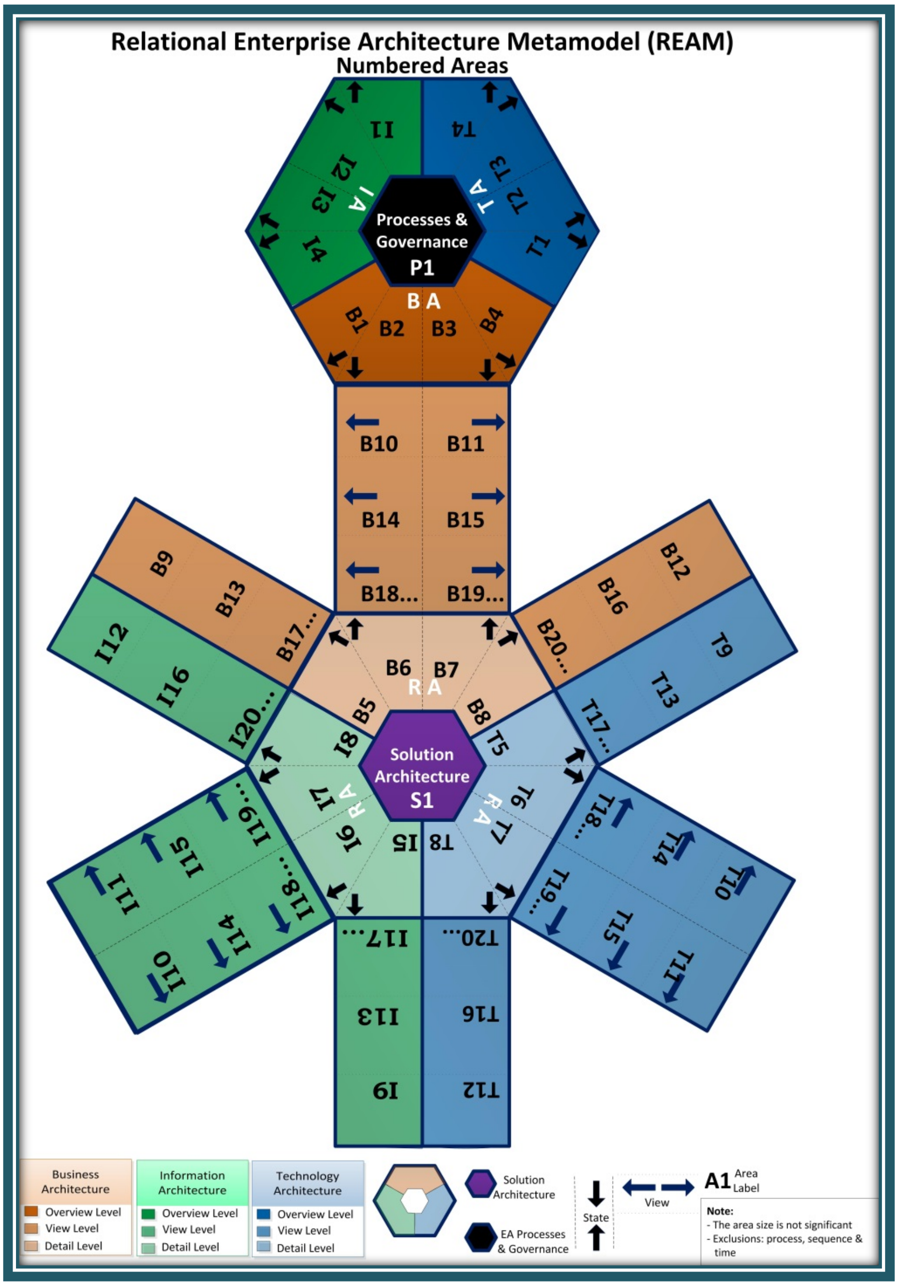


Figure 13-2: REAM with Changes to Numbering

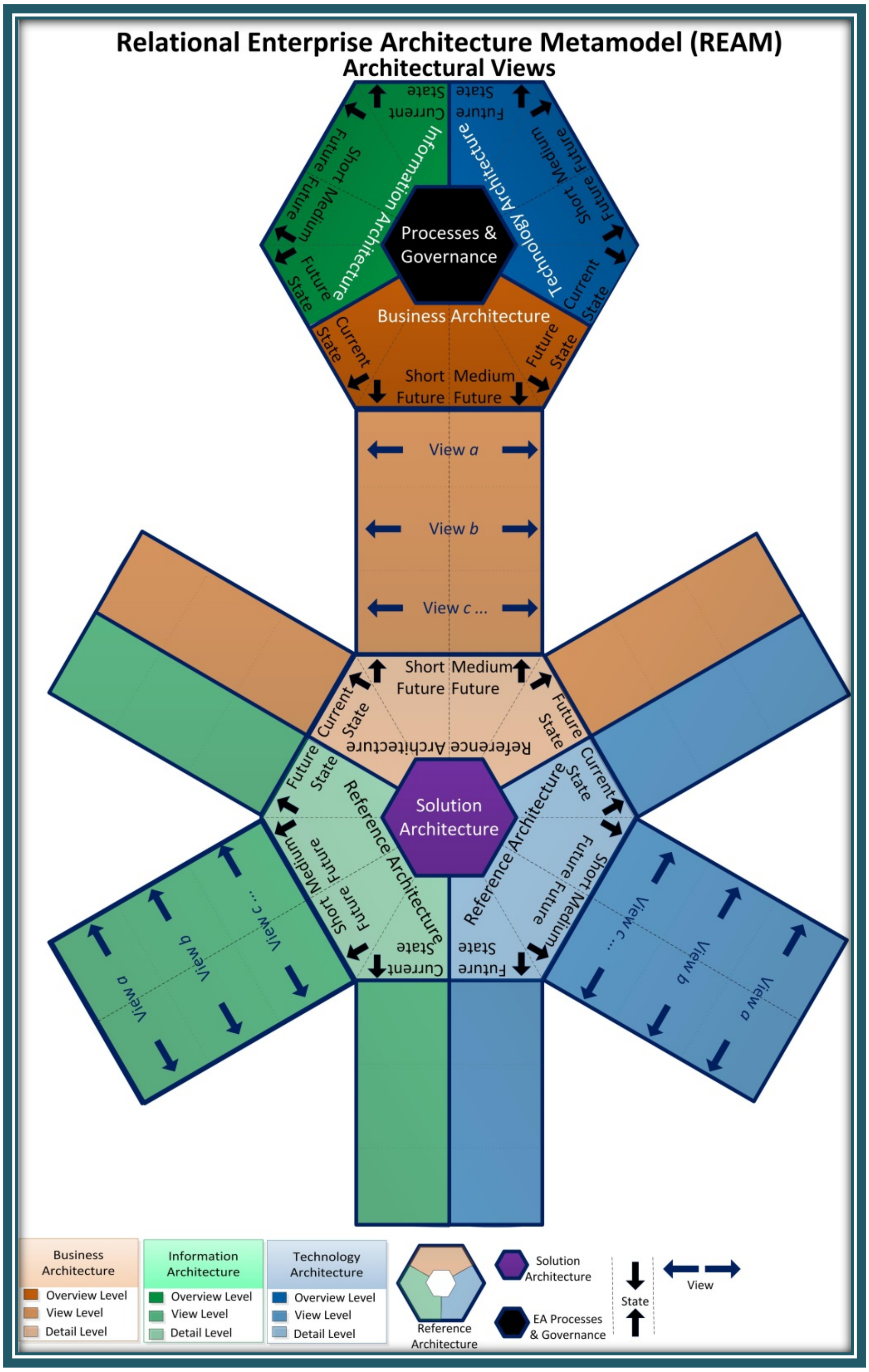


Figure 13-3: REAM with Changes to Views

13.3.1.3 <u>Visibility of Horizontal and Vertical Interlinking</u>

The visibility of the linkages, especially the vertical interlinking and traceability (Case Study A), should be enhanced and the possibility of creating horizontal and vertical slices (Case Study B) should be followed up.

The graphics of the horizontal linkages were changed to create more focus, while some vertical linkages were added to address this input. The shading of the colours depicts the vertical link between the levels of detail within the same architectural domain – this was emphasised by altering the legend. See Figure 13-4 for these changes, Figure 13-6 for the changed legend and Figure 13-7 for the composite Modified REAM.

The extraction of specific horizontal and vertical slices will have to be addressed more explicitly in a digital 3-D model.

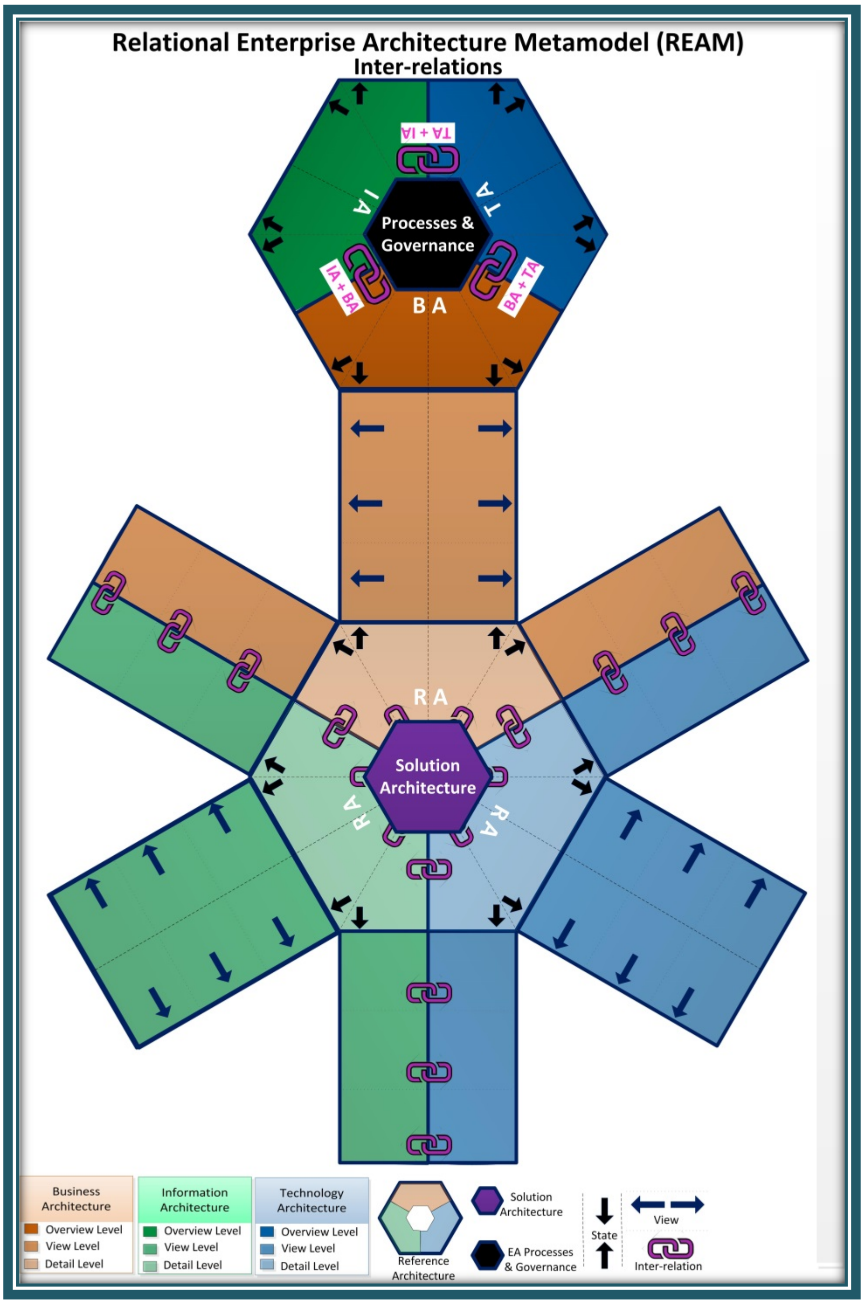


Figure 13-4: REAM with Changed Linkages

13.3.1.4 Detail Required

More detail or different levels of detail, for example, an object relationship view and detail of links were requested by two Case Studies (A and B).

This input is probably a result of the content and length of the contact sessions held with the Case Studies as per the research methodology. A brief overview (Addendum A:9.5) only was provided to the participants. More detail of the REAM is, however, available in paragraph 8.3.

13.3.1.5 Dynamic Size of the Areas

The depiction of the dynamic size of the areas was mentioned by Case Study B. Although the concept is that each area is flexible in size and is defined by the different components (colour, shade, arrows, links, *etc.*), the current depiction can create the impression that the size of the different areas is significant.

This input will be addressed by a note on the legend of the diagram – see Figure 13-5 and Figure 13-7 for the composite modified REAM.

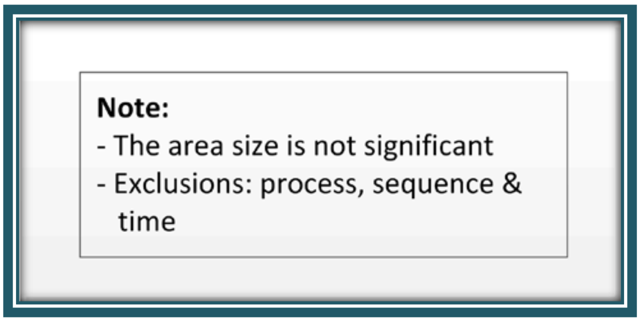


Figure 13-5: Legend Note

13.3.2 Clarification Inputs

Inputs, regarding items needing clarification, were received, evaluated and addressed.

13.3.2.1 Clarification of 'coherence'

The term 'coherence' as used to indicate the middle shade of each architectural domain needs clarification (Case Study B). The term was borrowed from a classification of viewpoints by Lankhorst *et al.* (2005: 163) and indicates the different levels of virtual scope.

This input will be addressed by changing the term to 'View Level' and altering the legend of the diagram – see Figure 13-6 for the changed legend and Figure 13-7 for the composite modified REAM.

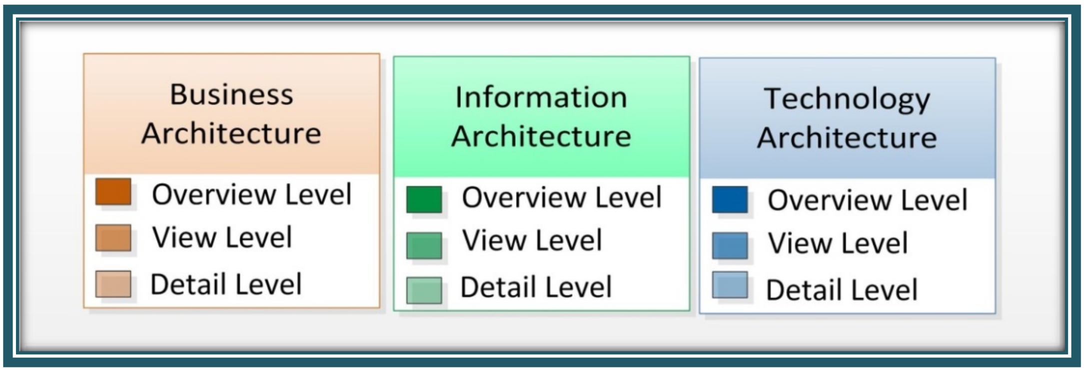


Figure 13-6: Changed Levels

13.3.2.2 The Purpose of the Metamodel

According to Case Study A, the purpose of the metamodel should be clear. This will assist in understanding where and for what purposes the REAM could be used within an enterprise. The purpose of the REAM is to provide a high-level metamodel, which encompasses and maps all EA architectural domains, with emphasis on the relationships and interlinking between them.

This input will be addressed by providing a subtitle: an interlinked EA map – see Figure 13-7.

13.3.2.3 Clarification on Excluded Dimensions

The dimensions not currently addressed (for example, time dimension, including iterations) should be acknowledged/mentioned/indicated in the REAM (Case Study A).

This input will be addressed by a note on the diagram – see Figure 13-5 for the legend note and Figure 13-7 for the composite modified REAM.

13.3.3 Additional Feature Inputs

The following inputs, regarding additional features, were received, evaluated and addressed.

13.3.3.1 Providing Context

All three Case Studies indicated the importance to provide context to the REAM, for example, a context diagram within the enterprise or even the extended context. For example, multiple instances within separate organisational units or companies within a group and relations with customers, partners and suppliers. The context could impact and influence the application of the REAM and the EA could influence the entities around it.

This input will be addressed by creating a context diagram. See the diagram in Figure 13-9.

13.3.3.2 Methodology

Two Case Studies (A and B) recommended adding a methodology or simple process guide and indicating the mandatory and/or optional areas of the REAM. The REAM is intended to be non-prescriptive regarding a specific methodology. As existing methodologies, such as ADM for TOGAF and FEAF, can be used as a methodology to apply and build the REAM, it is not deemed necessary to develop a separate methodology.

This input will thus not be addressed here as the REAM is intended to be methodology agnostic.

13.3.3.3 Additional Relationship Descriptors

Two inputs, both from Case Study C, were received regarding additional relationship descriptors for the linkages: add 'affected' and 'affected by' relationships; and add a security dimension to the linkages.

These inputs are valuable to provide more depth to the linkages. These will not be indicated on the metamodel diagram, but need to be incorporated in the detail levels of the REAM, by providing expanded or drill-down views. It can also become part of templates for the creation of artefacts,

13.3.3.4 RACI Role Indicators

It was suggested to add RACI role indicators to each area, by Case Study B which makes extensive use of RACI in all their processes and documentation.

This input could add value to the specific implementation of the REAM within an enterprise. On a high level it can indicate the different organisational units involved or responsible for a specific architectural area. On a lower level it can form part of the artefact templates and architecture processes. This will be implementation-specific and should not be prescribed by the metamodel as RACI indicates the roles within a process and not the description of a metamodel. The exact relation between the REAM and RACI can be the subject of future research.

13.3.4 Expansion Possibility Inputs

The following inputs were received for possible expansion of the REAM.

13.3.4.1 Physical and Digital Model

Provide a physical model of the REAM or build a 3-D REAM software model (Case Studies A and C). The need for a visual 3-D representation of the REAM was voiced. A physical model can assist in explaining, understanding and internalising the REAM. A digital model could be used to drill down into different deeper layers, containing more detail and examples.

These inputs are valid and a digital model will probably be the only feasible way of using and expanding the detail of the REAM. This could be addressed in a future project.

13.3.4.2 Fixed Views

One Case Study (C) proposed that the views in the REAM should be specific and fixed. This stems from the fact that this specific Case Study prepares all the architectural viewpoints for a solution upfront.

According to TOGAF (The Open Group, 2009c: 414) deciding on the viewpoints is an iterative processes conducted in different phases. Since the philosophy of the REAM is to be a flexible metamodel, which can be used in different ways and in different enterprises, the input to fix the viewpoints will not be incorporated at this point in time. Fixed views are applicable to an implemented model.

13.3.4.3 Example per Area

Case Study B suggested the inclusion of an example artefact in each area. This reminds of the Zachman ontology (Zachman, 2008) and the small example diagram in each cell.

This input could add value to the REAM and can be used in future to enhance it in an implemented model.

13.3.4.4 Comparison between REAM and TOGAF

A Case Study (A) suggested doing cross-checks or a comparison between the areas and dimensions of the REAM and the TOGAF metamodel. The result may provide insight into the comprehensiveness of terminology used in and differences between both frameworks/models.

This comparison could be considered for a future research project.

13.3.4.5 EA Maturity Evaluation

The last input suggests developing a maturity evaluation method based on the REAM (Case Study A). This stems from the clear mapping of the areas, which can be used to plot maturity.

This input could be considered for the subject of a future research project.

13.4 MODIFIED REAM

The composite REAM, as enhanced/modified according to inputs received from the Case Studies, is depicted in Figure 13-7. Although this is a complicated diagram, it depicts all the different dimensions simultaneously. The diagrams depicting only one dimension, such as Figure 13-1, Figure 13-2, Figure 13-3 and Figure 13-4, could be used as a simplified or focused view.

The updated table, describing the areas, is shown below as Table 13-1.

A context diagram of the REAM, indicating multiple instances and relevant context, is depicted in Figure 13-9. The depiction is based on the Rummler Business System diagram as described by Rummler and Brache (1995: 10) – see Figure 13-8.

Figure 13-7: The Composite Modified REAM

AREA	DESCRIPTION	AREA	DESCRIPTION
P1	Architecture processes and governance	I15	Medium-term future IA of View *b*
Business Architecture		I16	IA & BA part of View *b* & long-term future
B1	Overview of BA:IA relation & current state	I17 …	IA & TA part of View *c* … & current state
B2	Overview of BA short-term future state	I18 …	Short-term future IA of View *c …*
B3	Overview of BA medium-term future state	I19 …	Medium-term future IA of View *c …*
B4	Overview of BA:TA relation & long-term future	I20 …	IA & BA part of View *c* … & long-term future
B9	BA & IA part of View *a* & current state	I5	Detailed RA of IA:TA relation & current state
B10	Short-term future BA of View *a*	I6	Detailed RA of short-term future-state IA
B11	Medium-term future BA of View *a*	I7	Detailed RA of medium-term future-state IA
B12	BA & TA part of View *a* & long-term future	I8	Detailed RA of IA:BA relation & long-term future
B13	BA & IA part of View *b* & current state	**Technology Architecture**	
B14	Short-term future BA of View *b*	T1	Overview of TA:BA relation & current state
B15	Medium-term future BA of View *b*	T2	Overview of TA short-term future state
B16	BA & TA part of View *b* & long-term future	T3	Overview of TA medium-term future state
B17 …	BA & IA part of View *c* … & current state	T4	Overview of TA:IA relation & long-term future
B18 …	Short-term future BA of View *c …*	T9	TA & BA part of View *a* & current state
B19 …	Medium-term future BA of View *c …*	T10	Short-term future TA of View *a*
B20 …	BA & TA part of View *c* … & long-term future	T11	Medium-term future TA of View *a*
B5	Detailed RA of BA:IA relation & current state	T12	TA & IA part of View *a* & long-term future
B6	Detailed RA of short-term future-state BA	T13	TA & BA part of View *b* & current state
B7	Detailed RA of medium-term future-state BA	T14	Short-term future TA of View *b*
B8	Detailed RA of BA:TA relation & long-term future	T15	Medium term future TA of View *b*
Information Architecture		T16	TA & IA part of View *b* & long-term future
I1	Overview of IA:TA relation & current state	T17 …	TA & BA part of View *c…* & current state
I2	Overview of IA short-term future state	T18 …	Short-term future TA of View *c …*
I3	Overview of IA medium-term future state	T19 …	Medium-term future TA of View *c …*
I4	Overview of IA:BA relation & long-term future	T20 …	TA & IA part of View *c* … & long-term future
I9	IA & TA part of View *a* & current state	T5	Detailed RA of TA:BA relation & current state
I10	Short-term future IA of View *a*	T6	Detailed RA of short-term future-state TA
I11	Medium-term future IA of View *a*	T7	Detailed RA of medium-term future-state TA
I12	IA & BA part of View *a* & long-term future	T8	Detailed RA of TA:IA relation & long-term future
I13	IA & TA part of View *b* & current state		
I14	Short-term future IA of View *b*	S1	Solution Architecture

Table 13-1: Summary of Areas in the Modified REAM

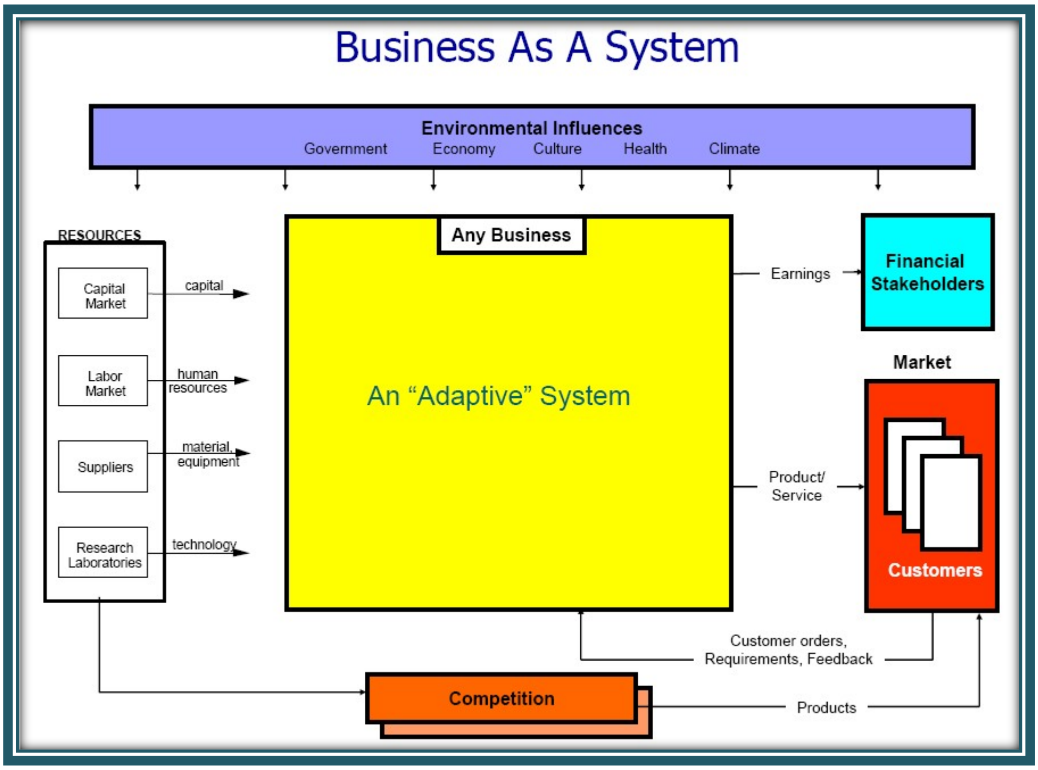


Figure 13-8: Rummler Business System (based on Rummler & Brache, 1995: 10)

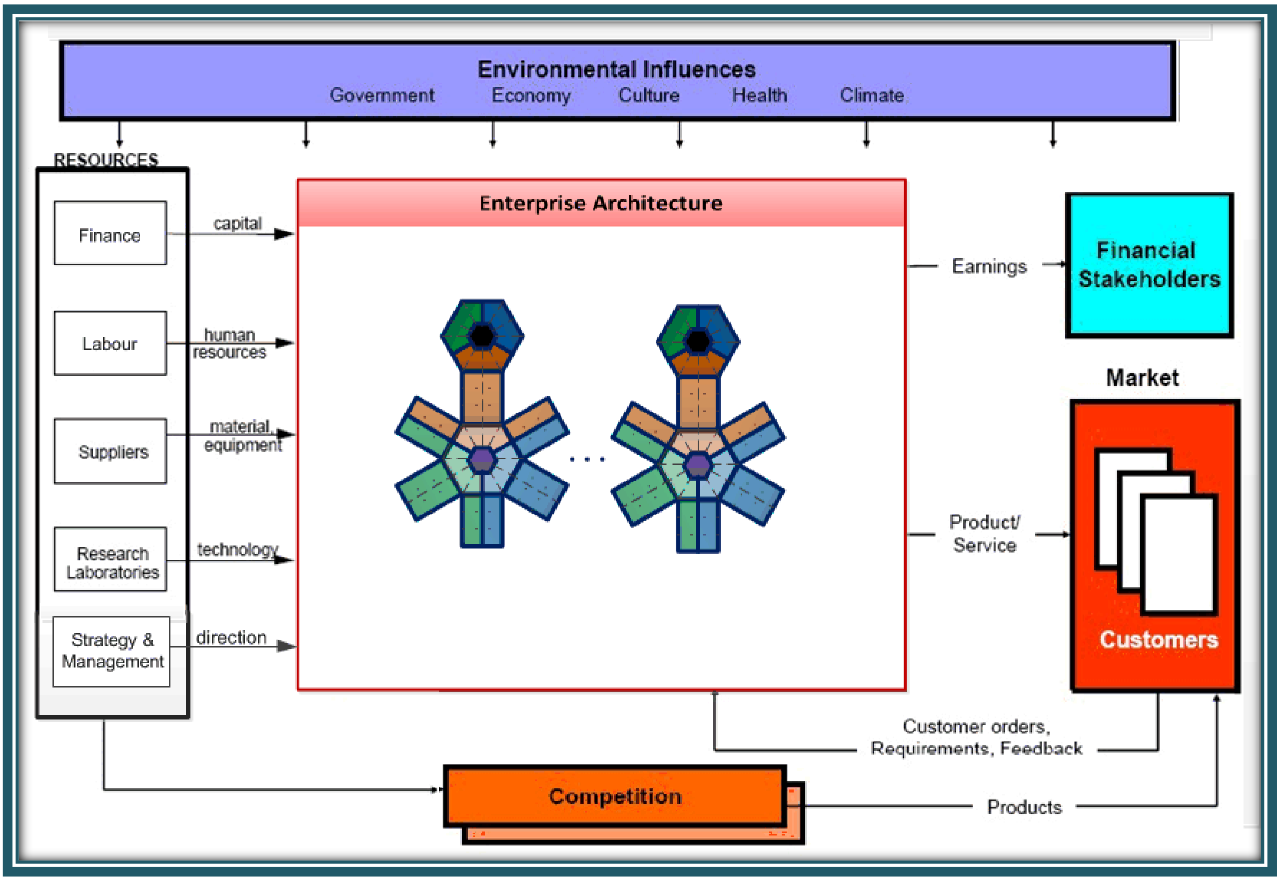


Figure 13-9: REAM(s) in a Context Diagram (adapted from Rummler & Brache, 1995: 10)

13.5 SUMMARY AND CONCLUSION

13.5.1 Summary

The sub-research question to be answered by this Chapter is:

*o) How was the proposed metamodel **modified**, based on the input from the case studies?*

All the enhancement/modification inputs from the Case Studies were combined in a list according to the categories: enhancements in depiction, clarification, additional features and expansion possibilities. Each input was addressed individually. This resulted in a modified REAM as depicted in Figure 13-7 and tabled in Table 13-1.

13.5.2 Conclusion

Since all three Case Studies accepted the REAM in principle, the inputs received were used to refine the REAM and not change it substantially. The valuable inputs were utilised to modify and enhance the REAM, without over-complicating it.

In the next Chapter the application possibilities of the REAM at the University of Pretoria will be investigated.

14 THE REAM AT THE UNIVERSITY OF PRETORIA

14.1 INTRODUCTION

In the previous Chapter the REAM was enhanced/modified based on the input gathered from the three case studies. The next sub-research question to be answered is:

p) *What is the applicability of the metamodel to the* ***University of Pretoria****?*

Initially, in order to determine the applicability of the REAM at the University of Pretoria (UP), the same research methodology as described in Chapter 9 was followed. The application was, however, done in more detail.

14.2 IDENTIFICATION OF ORGANISATION AND CONTACT PERSON(S)

The University of Pretoria was chosen in the educational vertical industry as this research is conducted for a degree at this university and because it is the researcher's place of work. It could also add value as their EA function is not as well developed as in the other case studies. The University of Pretoria is a South African-based public university. The University of Pretoria offers more than 1 800 academic programmes in two of the official languages, namely Afrikaans and English. In 1996, the University of Pretoria became the university with the highest overall research output in South Africa. The University of Pretoria celebrated its centenary in 2008. The academic programmes of the University are offered in nine faculties, as well as a business school. The faculties comprise a total of 140 academic departments and 85 centres, institutes and bureaus. In 2012 the University had ±63 000 students (±47 000 contact and ±16 000 distance students) and ±3 600 permanent employees (information gathered from UP's website[23]).

An appointment for an hour and a half was set up with the appropriate persons as the first contact session.

[23] For UP's website see http://www.up.ac.za [Accessed 15 May 2014].

14.3 FIRST CONTACT SESSION

A first contact session took place between this researcher and two contact persons (informants) from the University on 5 May 2014 for an hour and a half. The informants were the Director of Information Technology (within Information Technology Services) and the Manager: Integration. The informants signed the Informed Consent Form. Background on the research, based on Addendum A:14.1, was provided to the informants and a semi-structured interview, based on the questions in Addendum A:9.3 was, conducted. The following additional sample documentation was obtained by this researcher to complement the interview: a sample Project Start Architecture (PSA) (UPD1), Architectural Governance and Principles (UPD2), the EARB (Enterprise Architecture Review Board) Charter (UPD3), a Technology Planning Framework (UPD4), the ICT Master Plan (UPD5), a sample Technology Roadmap (UPD6) and two Architecture Maturity Assessments (UPD7). As with the Case Studies, these documents will be cited by the code in brackets and not bibliographically.

The interview question responses, supplemented by documents, are summarised as follows:

UP1:Q1 Please explain the enterprise architecture (EA) structure/set-up within your organisation with reference to business architecture, information architecture and technology architecture.

UP1:A1 Enterprise architecture is the ideal, but the IT organisation is too small to support a comprehensive EA function – the scope of EA is thus still limited. The EA function is relatively immature (UPD7) with a predominantly technology focus and is tactical and project-driven. EA is the responsibility of Information Technology Services. The time of one employee (currently the researcher) is dedicated to technology architecture, a little time from one other staff member (one informant) is allocated to application architecture and a little time from another (the other informant) is allocated to enterprise architecture.

Some business process modelling has been done, but in general no business architecture exists. Some information architecture is addressed as part of projects and a recently (March 2014) completed comprehensive systems renewal project. An Information and Data Governance Committee exists, setting the principles and parameters for information governance. Technology architecture is addressed by technology roadmaps, architecture diagrams, structured technology investigations and PSA's (see UP5:A1).

UP1:Q2 a) Do you make use of a recognised EA framework/model/methodology? b) If yes, how does this benefit your organisation, and c) how comprehensive is your implementation thereof?

UP1:A2 *No, a recognised EA framework is not used.* The choice of framework needs to be investigated and decided upon. The end result would probably be a hybrid or combination of frameworks/models.

UP1:Q3 a) How do you govern the EA function within your organisation? b) For example, do you have a formal exception process and what does it entail?

UP1:A3 An EA Forum is utilised for awareness, discussion of architectural issues and finalising certain architecture artefacts. The final approval of artefacts, principles and solutions is done by the Enterprise Architecture Review Board (EARB). The EARB Charter (UPD3) describes the scope, objectives, governance, roles and responsibilities and deliverables of the EARB.

The governance is described in the EARB Charter (UPD3) and the Architectural Governance and Principles (UPD2) document. Exceptions are handled as a formal request to the EARB.

UP1:Q4 a) Do you have a set of architecture principles that has been agreed upon? b) If yes, how and for what purpose do you use them?

UP1:A4 A set of architectural principles does exist – see UPD2.

The principles are reviewed yearly and are used as a starting point for architectural and solution decisions.

UP1:Q5 a) How is EA involved in your project management process? b) Do you make use of best practices/reference architectures or something similar?

UP1:A5 Architecture checks are incorporated in several stage gates of the project methodology of the Project Management Office. A Project Start Architecture is required for all infrastructure or technology projects. The PSA addresses business architecture (the business units, the products and services and the business processes changed/influenced/created), information architecture (applications changed/ influenced/created, authentication and data exchange), technology architecture (hardware, software, infrastructure, architecture diagram(s) and data centre impact) and architecture deviations (UPD1).

UP1:Q6 a) Does EA form part of other organisational and/or decision-making processes? b) If yes, which processes are they?
UP1:A6 EA provides inputs to the IT Human Resource allocation process and guides the university's centralised IT budget process. The ICT Master Plan (UPD5) provides potential input into the strategic planning process of the university.

UP1:Q7 What mechanisms do you use to ensure consistent integration/interlinking between the different architectural domains?
UP1:A7 No explicit mechanisms exist as yet. Currently interlinking is attempted within artefacts. For example, the technology roadmaps (UPD6) include references to IT business processes and the PSA (UPD1) has sections for all the architectural domains.

The implemented end-to-end Oracle architecture inherently supplies integration among systems. Integration documents are available, but not centrally accessible.

UP1:Q8 a) Is it necessary for you to distinguish between different views/viewpoints of EA to derive business value? b) If so, what are they?
UP1:A8 *It is definitely necessary for different views on an architecture.* Views will only be developed as and when required and useful, for example, for an executive proposal.

UP1:Q9 How do you address, for instance, security in your EA framework/model?
UP1:A9 Security is addressed in each technical team – no centralised security function exists. A Security Forum has recently been created and convened once. Security architecture is still immature.

UP1:Q10 How do you determine the architecture for the design and implementation of a new solution?

UP1:A10 The Technology Planning Framework (UPD4) guides the different channels for requesting technology solutions. New solutions are created as part of the project methodology. A PSA (UPD1) is created, based on the architectural principles, as the parameters for a solution. The set of PSA's and technology standards are the start of building a reference architecture. A solution can result in expanding the current architecture or applying for an exception through the EARB.

UP1:Q11 a) Do you have architecture artefacts such as depiction(s) or document(s) which explain your organisation's EA? b) If yes, are you making use of any EA artefact repository/application/tool? c) If so, what are they?

UP1:A11 A growing collection of architecture artefacts exists. The artefacts are registered at the ITS Document Control Centre and are stored on a collaboration space on Oracle WebCenter Spaces[24].

UP1:Q12 a) Do you document both the current and the future state of architecture? b) If so, where does the focus/emphasis lie?

UP1:A12 Currently the ratio is about 50:50 due to limited existing documentation. The plan is to work towards a 75% focus on the future and a 25% focus on current documentation.

14.4 ANALYSIS AND APPLICATION OF GATHERED DATA

The gathered data were organised and summarised in Table 14-1. The left-hand columns summarise the data and the two right-hand columns reflect the application of the proposed REAM for UP. This is used as input to the second contact session with UP.

[24] For WebCenter Spaces see http://docs.oracle.com/cd/E12839_01/webcenter.1111/e10147/Topic_4.1.htm [Accessed 7 May 2014].

	Q#	Question Subject	Summarised Answer	OU/Process	Example artefacts
Dimension : Architectural Domains				P1: Processes & Governance	
P	Q1	EA set-up	Small structure in ITS	Managed in IT Services	
	Q5a	EA in Project Management	Integral part of life cycle in projects. SVC	Architecture checks in stage gates	Project Start Architecture
	Q5b	Reference Architecture	Not formally, largely based on Oracle architecture	Reference architecture embedded in standards and PSA's	
	Q6a	EA in other processes	Inputs	Inputs to strategic planning, centralised IT budgeting and ITS HR planning	ICT Master Plan
	Q6b	List processes	Strategic Planning, budgeting and HR planning		
	Q3a	EA governance	EA Forum, EARB	EA Forum and EARB	Minutes, decision register
	Q3b	Exception handling	Motivation through EARB		
	Q4a	Architecture principles	Set of architectural principles reviewed annually		
	Q4b	Purpose of Principles	Architecture principles are used as a starting point for architecture and solution decisions	Architecture principles are used as a starting point for architecture and solution decisions	Architecture principles
				B1-B20: Business Architecture	
BA	Q1	EA set-up	No formal structure	No formal structure	Some business processes
				I1-I20: Information Architecture	
IA	Q1	EA set-up	No formal structure yet	No formal structure yet	Some data governance
				T1-T20: Technology Architecture	
TA	Q1	EA set-up	One person in IT	Managed in IT Services	Organogram
				S1: Solution Architecture	
SA	Q10	EA for solution	No formal structure	Solutions are handled through Technology Investigations and project methodology	Technology Planning Framework
Dimension : Architectural States				Architectural States	
	Q12	Document current/future	Equal effort, including transition states	Equal effort due to limited existing documentation	Technology Roadmaps

Dimension : Architectural Views				Architectural Views	
	Q8a	Distinguish views	Yes	Different viewpoints will be developed as required	Executive proposal
	Q8b	List views	-		
	Q9	Security	Security is handled within each team	Security is handled within each team	
Dimension : Interlinking				Interlinking	
	Q7	Integration/interlinking	Interlinking is included within artefacts	Interlinking is included within artefacts	
	Q5a	EA in PM	Project Methodology	Project Methodology	
	Q6	EA in other processes	Input into other organisational processes	Input into other organisational processes	
	Q11a	Artefacts	Yes		
	Q11b	Repository tool	Yes		
	Q11c	List	Oracle WebCenter Spaces	Oracle WebCenter Spaces store artefacts	
Background					
	Q2a	EA framework/model	No framework has been chosen		
	Q2b	Benefit	-		
	Q2c	Comprehensiveness	-		

Table 14-1: Data Analysis University of Pretoria

14.5 SECOND CONTACT SESSION

A second contact session was set up with the two informants on 12 May 2014 to evaluate the possible application of the REAM. The session was attended by the researcher and the two informants.

The session consisted of three parts.

First, the researcher delivered an introductory presentation, which explained the modified REAM and its different dimensions (see Addendum A:14.2). During the presentation on the REAM, the following were discussed:

- The possible differences or overlaps between a set of solution architectures and reference architecture. One possibility is to use a generic solution architecture as the reference architecture and create solution architectures per project.
- A view is a horizontal annulus or ring across all domains and states.
- A vertical slice focuses, for example, on a specific future state across all views and levels of detail.
- The numbering acts as coordinates to refer to an area. It can also be used as metadata elements in artefacts to indicate the scope of the artefact.

Second, the researcher delivered a presentation on the possible application of the REAM for UP (see Addendum A:14.3). The application of the REAM is based on the analysis of the information gathered during the first contact session. This presentation contained more detail than the similar presentations to the Case Studies, by drilling down into a specific area and discussing a sample artefact in that area. During this presentation the panel provided additional information and clarification on their EA activities. The essence of these additions is:

- Processes and Governance – architecture principles are used as a basis for the PSA.
- Business Architecture – Very limited in scope, with a possible next step the creation of an anchor model or a business capability model. The initiated development of a universitywide value chain could also assist business architecture.

- Information Architecture – a few artefacts do exist, for example, an application architecture map. Information architecture includes applications and information, in other words, information and the flow of information.
- Technology Architecture – a collection of artefacts exists.
- Architectural Views – the concept of views are useful, for example, business and technical. Some generic security practices are in place already.
- Interlinking – a next evolution of the roadmaps could place more emphasis on interlinking.
- Solution Architecture – guiding principles are to be taken into account here.
- Detailed application example – within the roadmap: the ITS view (swim lane) (*i.e.* **b** in Figure A:14.3-8) actually depicts the business processes view. The swim lanes regarding the resources and the required skills can be linked to the business (*i.e.* **a** in Figure A:14.3-8). The solution architecture can be a result of the PSA. The detailed explanation is *nice and very useful – it makes a lot of sense.*

Third, after the two presentations, the questions, as provided in Addendum A:9.6, were discussed. The interview responses and discussions are summarised as follows:

UP2:Q1 a) Please comment on the potential applicability of the REAM. b) What challenges do you foresee in the operational application of the REAM?

UP2:A1 The applicability and value of the REAM lies in *the provisioning of structure and order* to enterprise architecture and the flexibility to *implement in some areas only according to priorities beneficial to the organisation*. The flexibility of the methodology suits the university, because it has a very small EA function and less strict adherence requirements and more innovation than, for instance, a financial institution. The university needs a *light touch on architecture*. It can also prove very useful for a greenfield implementation, *i.e.* previously undeveloped sites.

The focus on interlinking can assist with the classification of artefacts, by referring to the area numbers. Even the links can be numbered for reference purposes. By adding the numbers to the metadata of the artefact, artefacts can be retrieved via a "*metadata taxonomy".* The links can explicitly be shown and implicitly be available through metadata. It is beneficial for the organisation if one can link the EA focus areas to the benefits for the organisation through alignment.

The Gartner concept of "pace layering" (Hotle & Kyle, 2012: 2) was also discussed in terms of how to link the concept of three differently governed system types (systems of records, systems of differentiation and systems of innovation) to enterprise architecture. Possibilities are: by using different views, classification in the metadata or separate solution architectures.

The REAM can have value as an instrument to convey the message of EA to other parts of the organisation and to indicate the importance of and the dependency on business architecture (including business processes). It was suggested that different views of the model be produced for different purposes, *i.e.* more abstract, less dimensions, *etc.*

UP2:Q2 Please comment on the comprehensiveness of the REAM (covering all aspects of EA).
UP2:A2 The REAM can accommodate all aspects that the informants are aware of. *Each aspect will find a potential place in one of the areas.*

UP2:Q3 Please comment on the ease of understanding and the ease of use of the REAM.
UP2:A3 Considering the fact that the informants are not primarily enterprise architects, they felt that there will be a learning curve to gain insight into the complexity of the metamodel. It was suggested elucidating the model with a few more examples in order to gain a better understanding. The metamodel would have been more complex if more architectural domains, such as security, were to be included separately.

UP2:Q4 a) The REAM provides explicit focus on interlinking. Does this provide value to the EA process? b) Does the application of the REAM improve the mutual influence between the architectural domains?
UP2:A4 *One purpose of interlinking is to ensure consistency through the different views* and another to test the correct interpretation of architecture through all the levels. *All the necessary links exist in the metamodel* – between domains as well as vertically.

The REAM improves the interlinking, because it is also a type of taxonomy, which can be linked through metadata. This will result in the capability to draw reports to correlate artefacts and to determine gaps, overlaps and even conflicting artefacts. Artefact templates could also improve interlinking. *The REAM offers a lot of potential*.

UP2:Q5 a) Could the metamodel contribute to improving the governance of EA? b) If yes, how could the metamodel facilitate the governance of EA?
UP2:A5 *Yes, in the sense of making issues explicit, but not directly.*

The REAM can, however, assist by indicating gaps and providing information as a structured basis for decision-making. *The metamodel is smart*.

UP2:Q6 What modifications to the REAM would you propose to improve EA in your organisation?
UP2:A6 The following possible modifications/enhancements were discussed:

- providing numbering for the links; and
- providing the ability to hide certain dimensions/details of the REAM, depending on the audience (although all the areas, data and links are still available in the background).
- The specification of metadata and keywords is part of the implementation and, correctly, should not be visible on the diagram.

14.6 SUMMARY AND CONCLUSION

14.6.1 <u>Summary</u>

The sub-research question to be answered by investigating the REAM at the University of Pretoria was:

p) *What is the applicability of the metamodel to the* ***University of Pretoria****?*

The research methodology and case study process (agreed upon), as defined in Chapter 9, was followed with UP in order to answer the above research question:

- Identify enterprise and contact person – the applicable contact persons within UP were identified and contacted.
- First contact session – a first contact session was conducted with two informants in the form of a semi-structured interview, based on the questions in Addendum 9-3, to gain background and understanding of the EA function at UP. The interview was complemented by relevant documentation from the enterprise. UP has a relatively immature EA function, very limited resources and centralised EA governance.
- Analysis and application – the EA operations of UP were analysed (see Table 14-1) and the REAM was applied to it (see Figure A:14.3-1 through Figure A:14.3-6 in Addendum A:14.3). A more detailed example was also presented (see Figure A:14.3-7 and Figure A:14.3-8 in Addendum A:14.3).
- Second contact session – a second contact session was conducted with the two informants. First, the modified REAM was presented to the panel. Second, the REAM, as applied to UP, was presented to the panel. Third, a semi-structured interview was conducted, based on the questions in Addendum A:9.6, to evaluate the REAM's applicability, comprehensiveness, ease of use, value of interlinking and governance.

UP views the **application possibilities** of the REAM to be the creation of structure and order, the flexibility of implementation scope and methodology, the enhancing of interlinking and to use as a tool to communicate about EA. The complete composite diagram might be a bit too complex. Possible enhancement of the REAM could include the numbering of the links and the provisioning of different views of the REAM depending on the audience. The possibilities of achieving a synergy between, for example, the roadmaps or pace layering, and the REAM were discussed – this can be addressed in the future.

14.6.2 Conclusion

Based on the above outcomes, it can be concluded that UP has a relatively immature and still very small EA function. The REAM was understood as a comprehensive metamodel, which could contribute to EA decision-making, could provide structure and order and could interlink artefacts and solutions. Emphasis was placed on the possibilities of using the REAM's area (and link) numbering as metadata for EA artefacts to enhance visibility, traceability and interlinking. A couple of possible enhancements was gained from UP.

The next and last Chapter will contain the summary and conclusions of the study and indicate future related research areas.

15 SUMMARY AND CONCLUSIONS

15.1 INTRODUCTION

This research is completed by summarising the findings and methodologies and providing recommendations and concluding remarks.

The detailed findings pertaining to the specific research questions are reported below, followed by a summary of the broader findings (conclusions). The research is evaluated by reporting on the research methodology and explaining the value and contribution of this research. Recommendations stemming from the research are made and possible future research projects are suggested.

15.2 RESEARCH FINDINGS

The research findings will be reported by describing the detailed findings and the broader findings. According to Badenhorst (2008: 206) the detailed findings (referred to as close findings by Badenhorst) are directly inferred from the analysis and is related to the data collected. The broader findings provide a synthesis of the research findings and relate the detailed findings to the original research question.

15.2.1 Detailed Findings

The detailed findings of this research are the responses to the problem statement as represented by the sub-research questions.

The main research question is:

> *What should the key characteristics be of a metamodel for enterprise architecture, which focuses on the interfaces between the different architectural domains?*

The main research question was subdivided into three sections and nineteen sub-research questions: a) through s). The sub-research questions will each be listed below, with their findings.

The first section entailed a critical analysis of the existing literature and addressed the following sub-research questions:

*a) What definitions, frameworks and models are there for **enterprise architecture**?*

This question was answered, in Chapter 2 paragraphs 2.2 and 2.3, by exploring a representative sample of the definitions, the history, the existing frameworks and the existing models of enterprise architecture. Furthermore, the different domains, which form part of enterprise architecture, were investigated.

- Several **definitions** from literature were cited and analysed. An synthesised definition was compiled (Figure 2-1), is repeated here in Figure 15-1 and is written out as:

  Enterprise architecture is the process of describing, modelling, communicating, applying and governing the strategies, processes, current state, future-state blueprints, interrelationships, change/innovation and alignment/integration of the business, information, technology and information systems of an enterprise.

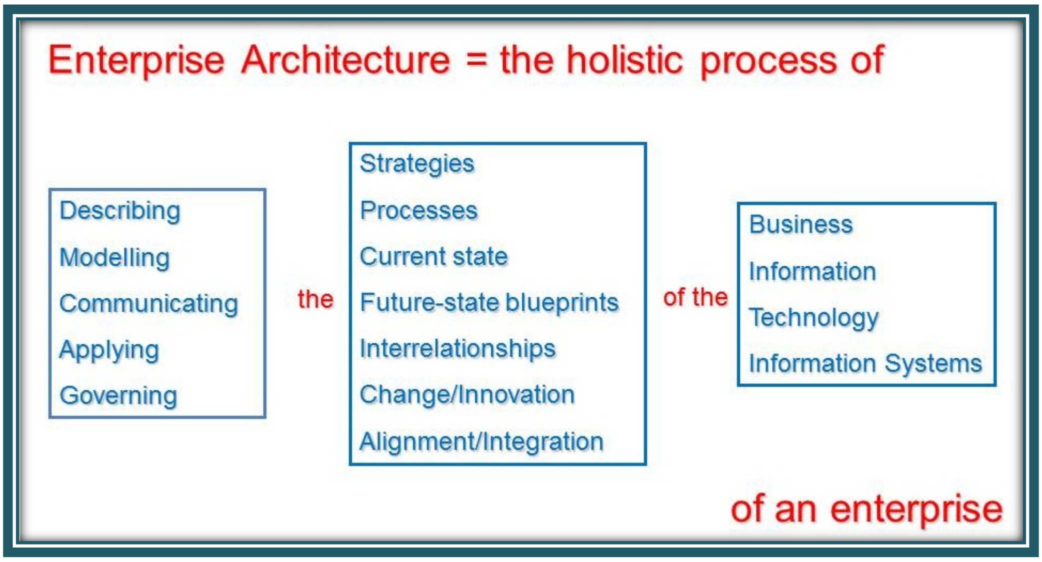


Figure 15-1: Synthesised Definition of Enterprise Architecture

- The **history** was described from 1987 till 2010 in terms of frameworks and their interdependencies. The foreseen future was described based on Gartner's hype cycle (Burton & Allega, 2010).
- The concept of a **framework** was described. The following frameworks were each described in terms of overview, scope, views, abstractions, the system development

life cycle, strengths and weaknesses: the Zachman Framework, TOGAF, DoDAF and FEAF. A few other frameworks, *i.e.* E2AF, SAGA and GERAM, were briefly discussed.

- In the literature, the use and interpretation of the term **model** within the enterprise architecture context is not consistent. Generic models can, however, be utilised to describe or design specific artefacts as part of the enterprise architecture process. A short description of the Process Model, the Relational Model, the Causal Loop Diagram Model and the Object-Oriented Model was provided.
- The main architectural **domains** of enterprise architecture are business architecture, information architecture and technology architecture and should be depicted in a high-level core diagram for managers and executives.

b) What are the rationale, purpose and role of enterprise architecture?

This sub-research question was answered in Chapter 2, paragraphs 2.4, 2.5 and 2.6.2. The threefold **role** of enterprise architecture within an enterprise is summarised in Table 2-2 and is repeated as Table 15-1, below.

ROLES	***DESCRIPTION***
Vision	Gaining a holistic view and vision of the future state of the enterprise.
Documented Complexity	Capturing the complexity of the enterprise in a manageable fashion.
Framework	A framework and toolset for implementation and application of enterprise architecture.

Table 15-1: Roles of Enterprise Architecture

The main **functions** of enterprise architecture have been described. These functions can be converted into multiple **benefits** for an enterprise by documenting enterprise architecture effectively and efficiently. A summary of the findings is tabled in Table 2-3 and is repeated below in Table 15-2.

FUNCTION	DESCRIPTION OF BENEFIT
Alignment	Enabling alignment across all the levels and domains of an enterprise.
Information Provisioning	Provisioning of stable and consistent information on the whole enterprise. Information provisioning for decision support. Providing information and alternatives as a planning tool. Satisfying customer requirements.
Driving Innovation	Insight into and a vehicle for driving innovation and change.
Creating Agility	Enhance responsiveness and agility within the enterprise.
Utilisation of Resources	Improved effective and efficient utilisation of IT resources. Reduction of IT costs. Reduction of duplication.
Process Enhancement	Improved communication processes. Assistance in strategic governance. Optimisation of internal operations and business processes. Consistent and integrated enterprise architecture processes.
Risk Reduction	Increased risk tolerance and insight.
External Linkage	Increased flexibility and knowledge to enhance linking with external partners.

Table 15-2: Benefits of Enterprise Architecture

*c) What are the benefits and challenges in **documenting** enterprise architecture?*

Documenting the enterprise's enterprise architecture (Chapter 2, paragraph 2.6.1) is a vital component of the architecture process and should focus on the future state, with only sufficient emphasis on the current state. Every function (described in sub-research question *b)*) will result in a benefit if fulfilled successfully and will not be repeated here. The additional **benefits** of documenting enterprise architecture are tabled in Table 2-4 (repeated here as Table 15-3).

BENEFIT	DESCRIPTION
Execution Foundation	Building a foundation for execution because it maps out processes, data and technology.
Intellectual Capital	Capturing and retaining the existing wealth of intellectual capital in the enterprise.
Value Realisation	A value realisation process, running concurrently with the documenting process, will make the value more visible to the enterprise.
Measurables	Providing metrics to enterprise architecture

Table 15-3: Benefits of Documenting Enterprise Architecture

Documenting an enterprise's enterprise architecture presents its fair share of **challenges**, which are summarised in Table 2-5 (repeated here as Table 15-4).

CHALLENGES	DESCRIPTION
Double-barrelled approach	To balance out the increase of IT efficiency while continuing business innovation or in other words complicated systems and complex systems.
Scalability	To create scalable architecture that has the ability to change.
Measuring	To acquire the assessment tools to prove value to enterprise.
Ambiguous documentation	To avoid ambiguous documentation by different stakeholders or sections of the enterprise.
Rate of Change	To deal with the rate of technology and business changes
Maintenance	Maintenance of the models and documentation
Co-operation of enterprise	Commitment and current state of enterprise can impede the architecture process.
Communication	Communication and promotion of enterprise architecture within the enterprise.
Governance	Governance of the enterprise architecture

Table 15-4: Challenges in Documenting Enterprise Architecture

*d) What definitions, benefits and models are there for **business architecture**?*

In Chapter 3 this question was addressed by first evaluating existing definitions of business architecture and then compiling a working definition thereof. The synthesised working **definition** is depicted in Figure 3-3, is duplicated here in Figure 15-2 and is written out as:

> *Business architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships and change/innovation of the business strategies/objectives, processes/value chains, capabilities, functions/structure and resources (human and finance) of an enterprise.*

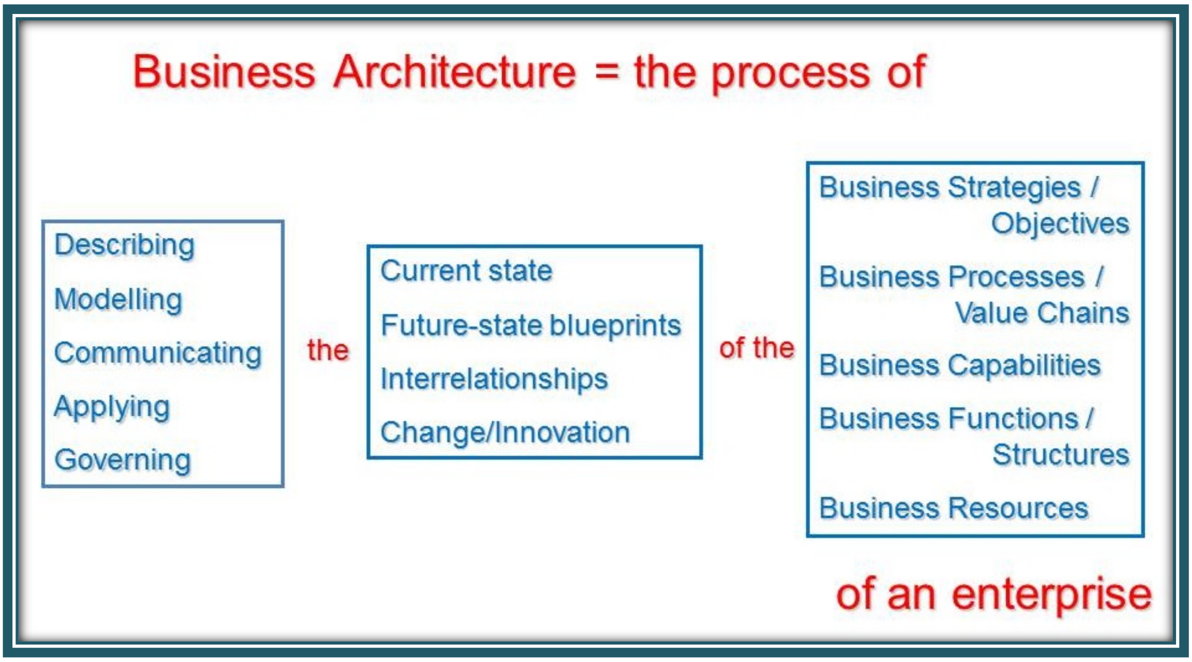


Figure 15-2: Synthesised Definition of Business Architecture

Second, facets of the **role** are, in short, flexibility in change, organisation, creation of business artefacts and gaining insight into and understanding of the business. The **benefits** lie in business architecture's assistance in structuring, guiding, shaping, managing and improving the enterprise.

Third, the different levels of abstraction with regard to **frameworks** and **models** were discussed. Table 3-2 provides a quick overview of the abstraction levels and is repeated here as Table 15-5.

TYPES	*EXAMPLES*
EA Frameworks	Business Concepts (Zachman Framework) Business Architecture (TOGAF) DoDAF Business Reference Model (FEAF) Gartner Business Architecture Framework
BA Frameworks	Microsoft Services Business Architecture Methodology A New Business Architecture for UC Agile Business Process Modelling Framework Business Motivation Model
Modelling Languages	Process Model Relational Model Causal Loop Diagram Object-Oriented Model Pi-Calculus Process Algebra Business Process Model and Notation Capability Model Web Services Description Language (WSDL) Business Process Execution Language (BPEL) Semantics of Business Vocabulary and Business Rules (SBVR) Archimate
Ontologies	Fact-Based Ontologies DDPO (DOLCE + DnS Plan Ontology)

Table 15-5: Overview of Levels of Frameworks and Models (Business Architecture)

*e) What definitions, benefits and models are there for **information architecture**?*

In Chapter 4, this question was addressed by first evaluating existing definitions of information architecture and then compiling a working definition thereof. The synthesised working **definition** is depicted in Figure 4-1, is duplicated here in Figure 15-3 for ease of reference and is written out as:

> *Information architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships, change/innovation, usability and sharing of the information assets/content, information activities, and the information audience of an enterprise.*

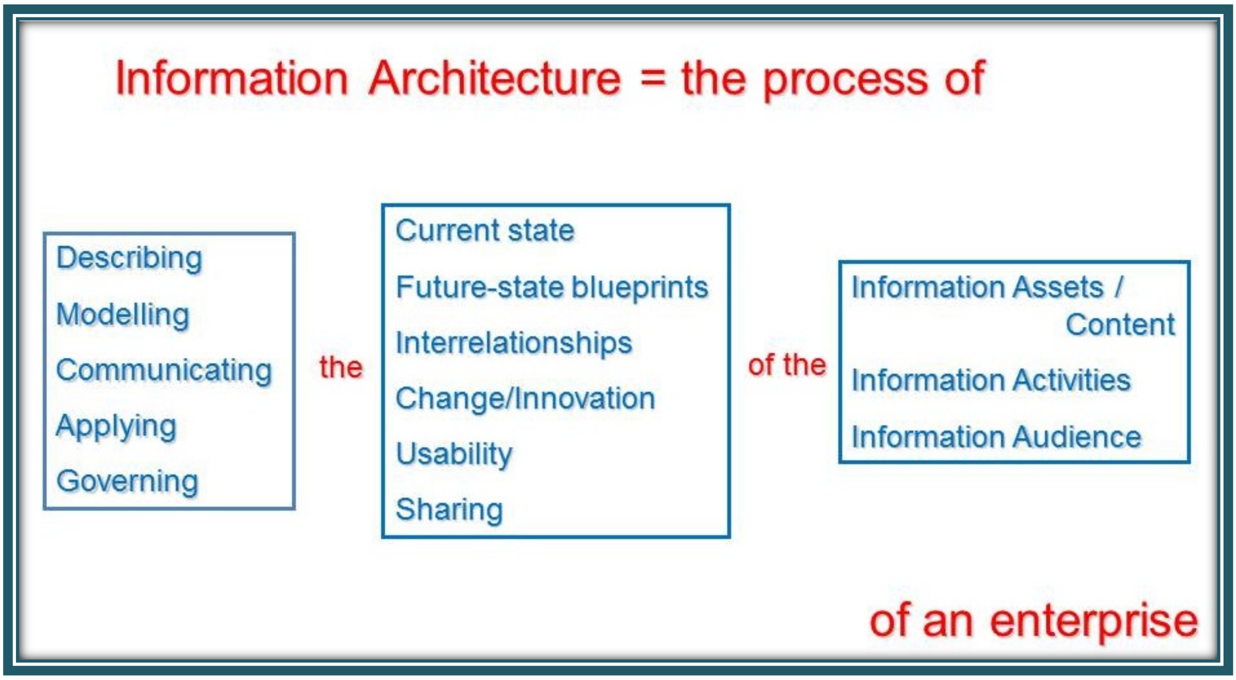


Figure 15-3: Synthesised Definition of Information Architecture

Second, facets of the **role,** within an enterprise, are to manage its information assets by using common languages/structures/classifications to enhance integration, solutions, cost reduction, access, task completion and competitive advantage. The **benefits** lie in information architecture's assistance in improving access to information (including sales and brand loyalty), reducing duplication of effort and enhanced communication between business and IT.

Third, the different levels of abstraction and examples with regards to **frameworks** and **models** were discussed. Table 4-2 provides a quick overview of the abstraction levels and is repeated here as Table 15-6.

TYPES	*EXAMPLES*
EA Frameworks	Systems Concepts (Zachman Framework) Information Systems Architecture (TOGAF) Systems & Services View (DoDAF) Data Reference Model (FEAF) Gartner Information Architecture Framework
IA Frameworks	Services Orientated Architectures 'Facets are Fundamental' Framework The Common Knowledge Enterprise Model Information Architecture Abstract Model Strategic Information Architecture The Evernden Eight
Models	Metadata-Modelling Information Interactive Model Generic Model for EIA for a Public Institution Instructional Design Model Usability Model Conceptual Model Extended Influence Diagrams

Table 15-6: Overview of Levels of Frameworks and Models (Information Architecture)

There is also a variety of descriptive languages available, most of which are related to XML, for example, XSD, DITA, SOX and SXML.

f) *What definitions, benefits and models are there for documenting* ***technology architecture*** *or technical architecture?*

In Chapter 5, this question was addressed by first evaluating existing definitions of technology architecture and then compiling a working definition thereof. The synthesised working **definition** is depicted in Figure 5-1, is duplicated here in Figure 15-4 for ease of reference and is written out as:

Technology architecture is the process of describing, modelling, communicating, applying, governing and maintaining the current state, future-state blueprints, *interrelationships and change/innovation of the IT systems, infrastructure, strategy, portfolio and investment of an enterprise.*

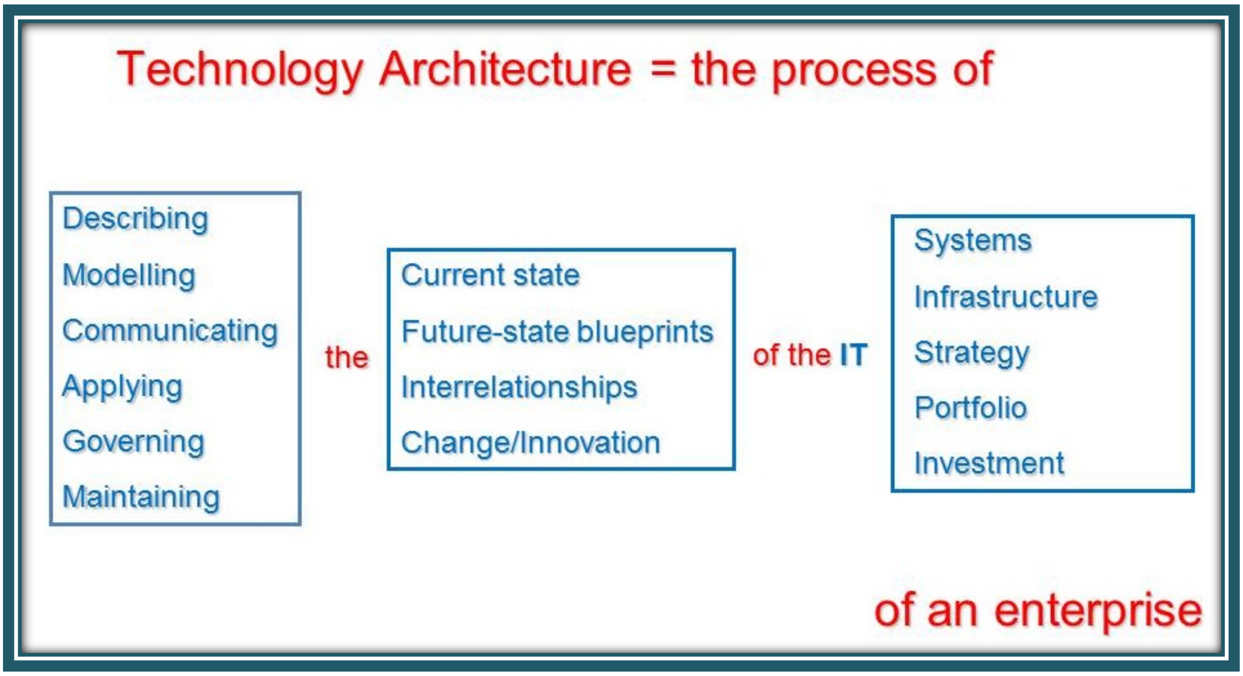


Figure 15-4: Synthesised Definition of Technology Architecture

Second, facets of technology architecture's **role** within the enterprise are, in short, to govern IT investments and assets, describing the IT plans, infrastructure and interrelations, and enabling innovation and flexibility. Technology architecture can, in a nutshell, produce **benefits** by reducing cost, improving decision-making, operational effectiveness and business alignment and providing growth in flexibility.

Third, the different levels of abstraction with regard to **frameworks** and **models** were discussed. Table 5-2 provides a quick overview of the abstraction levels and is repeated here as Table 15-7.

TYPES	*EXAMPLES*
EA Frameworks	Technology, Components & Operations (Zachman Framework) Technology Architecture (TOGAF) Technical Standards View (DoDAF) Technical Reference Model (FEAF) Gartner IT Architecture Guideline Framework
TA Frameworks	IT Framework (Carbone) Strategic Technology Architecture Roadmap Technology Architecture Framework (Victoria University)
Models	Data Flow Model MEGA Architecture CORA Model

Table 15-7: Overview of Levels of Frameworks and Models (Technology Architecture)

*g) What are the **taxonomy** and **relationships** of the different architectural domains?*

This question was answered, in paragraph 6.2, by

- providing an overview of the literature – it was evident that there is a variety of different depictions of enterprise architecture (six examples were discussed);
- subsequently proposing a taxonomy – see Table 6-1 (repeated here as Table 15-8); and
- proposing an integrated depiction of enterprise architecture – see Figure 6-8 (repeated here as Figure 15-5).

Governance (inlcuding performance and exceptions)

Enterprise Architecture

- (Enterprise) Business Architecture
  - Capability Architecture
  - (Business) Process Architecture
  - (Logical) Service Architecture
  - Product Architecture
- (Enterprise) Information Architecture
  - Information Security Architecture
  - Knowledge Architecture
  - Integration Architecture
  - Information System Architecture
    - System(s) Architecture
    - Application Architecture
  - Data Architecture
    - Data Delivery Architecture
- (Enterprise) Technology Architecture
  - Technology Security Architecture
  - Software Architecture
    - Event Architecture
  - Middleware Architecture
    - Database Architecture
    - Web Architecture
    - Service-Oriented Architecture
  - Infrastructure architecture
    - Network Architecture
    - Server Architecture
    - Platform Architecture
    - Communications Architecture
  - Hardware Architecture

(Enterprise) Solutions Architecture

Reference Architecture

Architecture Governance (including performance and exceptions)

Governance (inlcuding performance and exceptions)

Table 15-8: Proposed Taxonomy for Enterprise Architecture

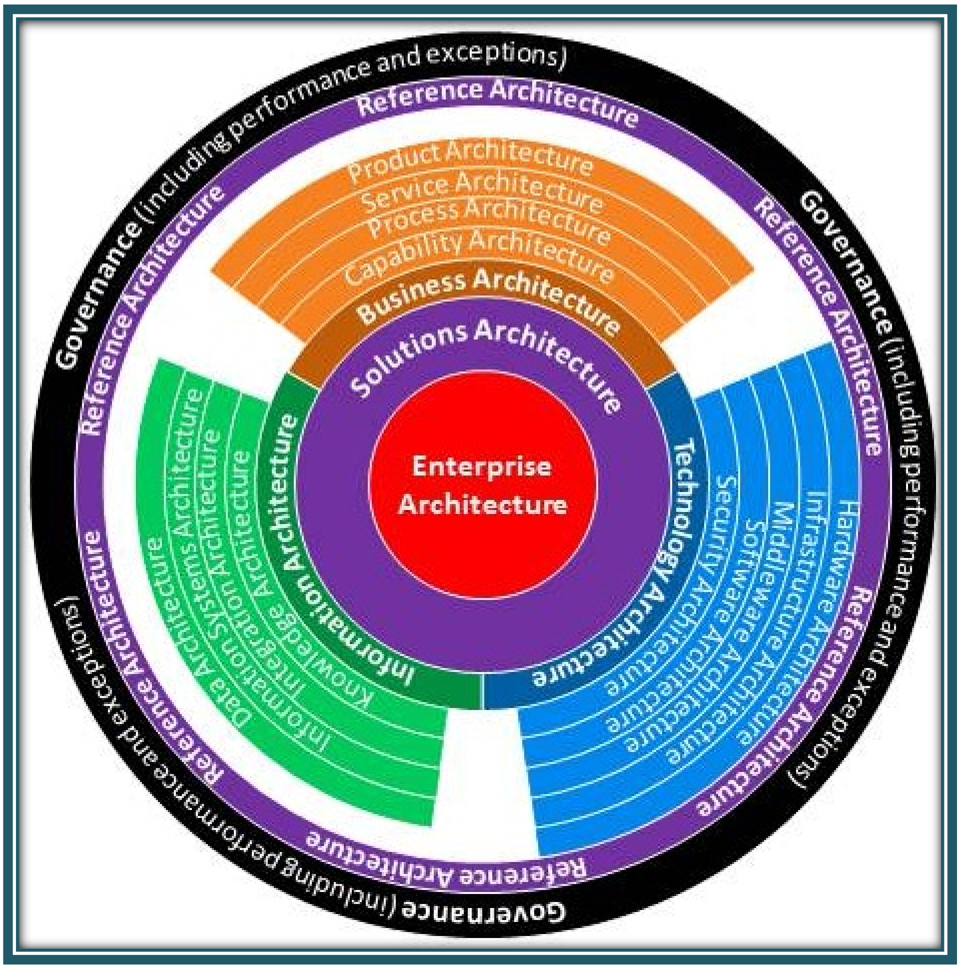


Figure 15-5: Proposed Integrated Depiction of EA

*h) What definitions, benefits and models are there for **integrating**, interacting and/or interlinking the architectural domains or are utilised for indicating the relationships between the architectural domains?*

This question was addressed in Chapter 6 and answered by

- providing an overview of the definitions in literature – in the absence of specific definitions, the different **terms** used in the literature were investigated and placed on a relationship continuum – see Figure 6-10 (repeated here as Figure 15-6). The terms 'alignment', 'relations' and 'integration' have the highest occurrence. The terms *'relation'*, '*integration strategy'* and '*interlinking mechanisms'* were chosen to be used (refer to paragraph 6.3).

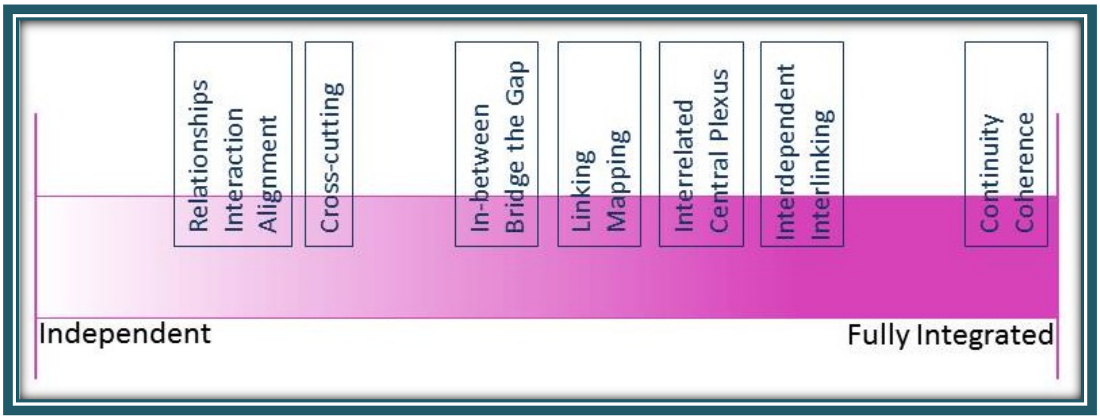


Figure 15-6: Relationship Scale of Terms

- providing an synthesised **definition** in paragraph 6.3 – see Figure 6-12 (repeated here as see Figure 15-7):

  Relating architectural domains is the integration strategies and the interlinking mechanisms used to combine two or more of business architecture, information architecture and technology architecture to provide meaningfully aligned enterprise architecture, in the context of an enterprise.

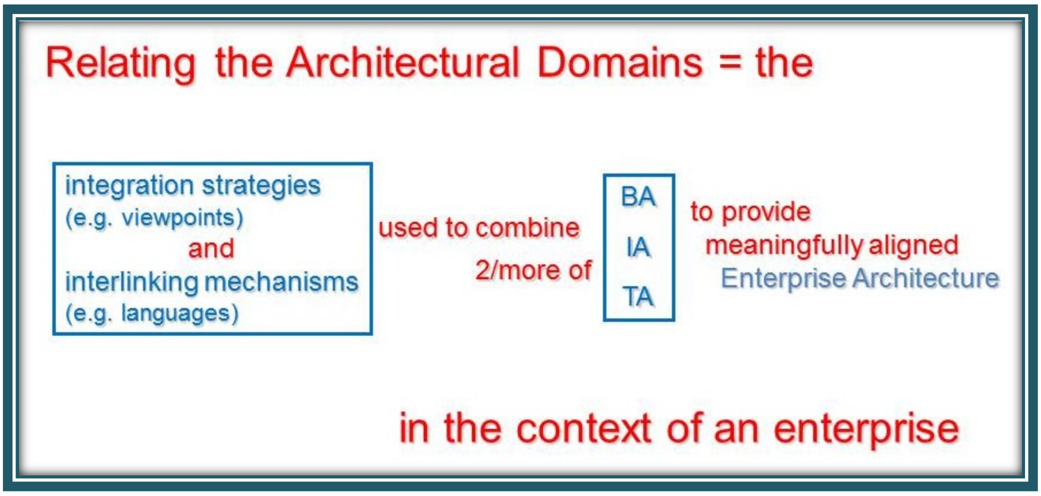


Figure 15-7: Synthesised definition of Relating of Architectural Domains

- determining the **role** of relating the architectural domains – clear, global, co-ordinated, unambiguous modelling to manage complexities and enable innovation (refer to paragraph 6.4).
- determining the **benefits** of relating the architectural domains – enabling the innovation/change and business processes, flexibility and the re-use/sharing of a variety of objects (refer to paragraph 6.4).
- investigating the relationships between the architectural domains in the main EA **frameworks** (paragraph 6.5.1) – see Table 6-3 (repeated here as Table 15-9 below)

for an overview. It was found that the relations of EA domains consist of two components, being the integration strategy (what to relate) and the interlinking mechanisms (how to relate):

- describing the **integration strategies** and **interlinking mechanisms** used by other frameworks (paragraph 6.5.2) – see Table 6-3 for an overview (repeated here as Table 15-9).

FRAMEWORKS	*INTEGRATION STRATEGY*	*INTERLINKING MECHANISM*
EA FRAMEWORKS		
Zachman	2D-Matrix	Human intervention and alignment of columns
Zachman DNA	Multiple 3D-Matrix	Integrated activities in third dimension
Zachman (Graves)	Multiple 3D-Matrix	Built-in processes in third dimension
TOGAF	Metamodel (standard taxonomy, reference architecture)	Common Language
TOGAF (Gerber, Kotzé & Van der Merwe)	Metamodel and Ontologies	Common Language & Ontologies
DoDAF	Views, metadata and taxonomy	Net-centric common language
FEA	Metamodel (reference architecture)	Architecture processes (transitional)
GEA	Meta-architecture – solution architecture	Process model
INTEGRATION FRAMEWORKS		
Domain Architecture	Viewpoints	Partitioning (inter-domain links)
Service Orientation	Service layers	Relationships (*e.g.* used by)
Viewpoints	Viewpoints	Iterative stakeholder processes
Metamodels	Metamodel	Common language & templates
EA Framework (Rohloff)	Views	Blueprints
CEiSAR	3D-Cube	Synergy split

Table 15-9: Summary of Integration Strategies and Interlinking Mechanisms

- listing a few examples of **models/methods** for creating integration content, such as delineation, interoperability, modularity, capability mapping, value chain analysis, business engineering navigator, component business model, activity model and process model (refer to paragraph 6.5.3).

The second section entailed the modelling methodology and process followed to create a metamodel. The metamodel needs to provide structure as well as a solution to documenting the relations between the different architectural domains. The following sub-research questions were addressed:

i) *What will the **modelling** process look like, including the elements and deliverables which need to be addressed?*

The first part of this sub-research question was answered by outlining the iterative modelling **process** (refer to paragraph 7.2). The process consists of:

- Defining the problem – understand, formulate and scope the problem;
- Background research – orientate, analyse environment, specify requirements, define specifications and provide context;
- Design/development – create, evaluate and consider alternatives and then model, create mock-up and/or prototype and produce the metamodel;
- Evaluation and refining – evaluate, test and refine the metamodel;
- Completion – launch the product / implement the system / sign-off the project.

The second part of this sub-research question was answered by compiling and discussing a list of modelling **elements** and **deliverables** (refer to paragraph 7.3). The metamodel should contain written descriptions of all its elements and deliverables. The resulting list of modelling elements and deliverables is as follows:

- Applicable standards
- Documentation
- Design principles
- Definition of terms
- Depictions
- Model description

- Viewpoint specifications
- Architectural views
- Alignment approach
- Stakeholders and concerns
- Methodologies.

*j) How will an integrated metamodel, interlinking the different architectural domains, **be constructed** and described?*

Five actions were taken to answer this sub-research question. First, the **re-usable concepts**, from the explored existing frameworks, were extracted (paragraph 8.2.1). The six significant ideas from existing frameworks, *i.e.* views, TOGAF processes, solution architecture, viewpoint classification, MOF processes and cube representation, as well as the other two useful artefacts, *i.e.* taxonomy and depiction, were taken into account to create and evaluate design alternatives.

Second, three possible **design alternatives** were developed and evaluated (paragraph 8.2.2):

- Design Alternative A was based on the previously proposed EA depiction – see Figure 8-2 (repeated here as Figure 15-8);

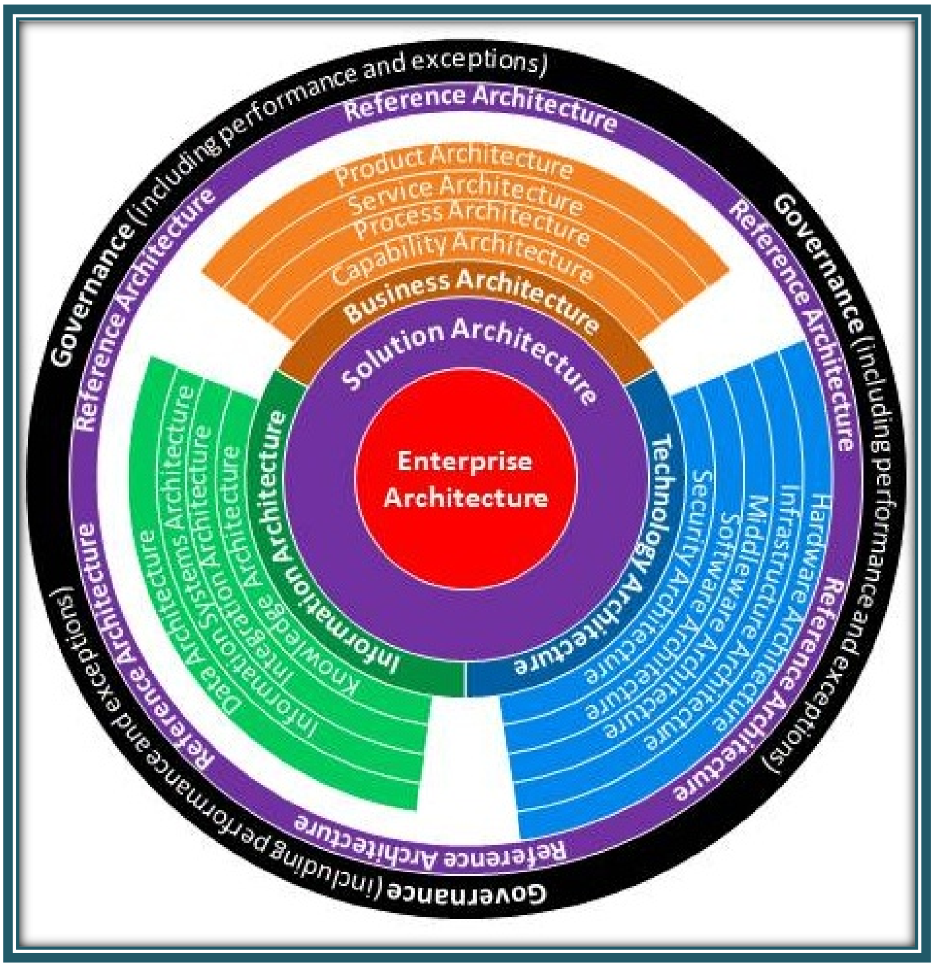


Figure 15-8: Design Alternative A

- Design Alternative B was based on Venn diagram principles – see Figure 8-3 (repeated here as Figure 15-9); and

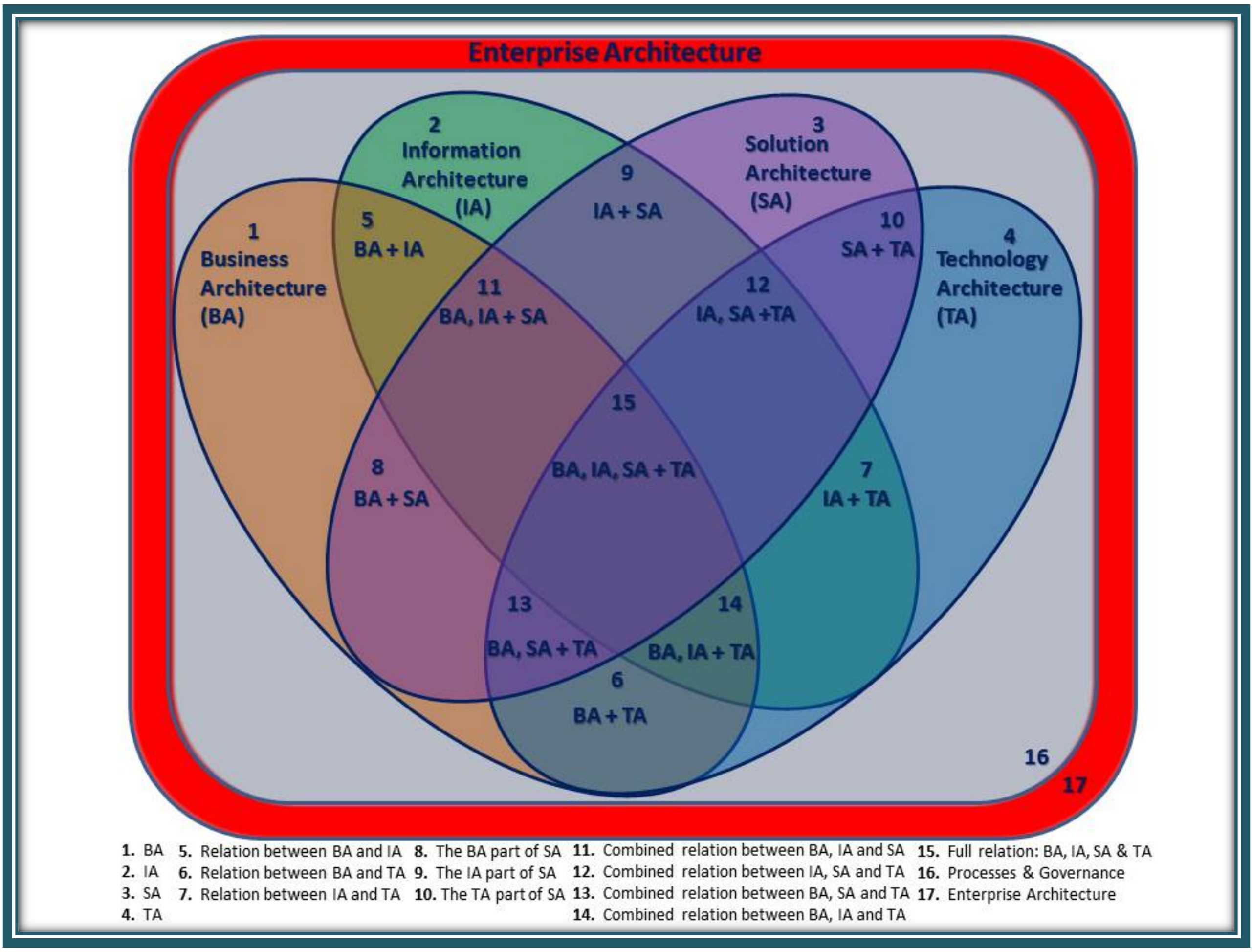


Figure 15-9: Design Alternative B

- Design Alternative C was based on the CEiSAR cube principles – see Figure 8-4 (repeated here as Figure 15-10).

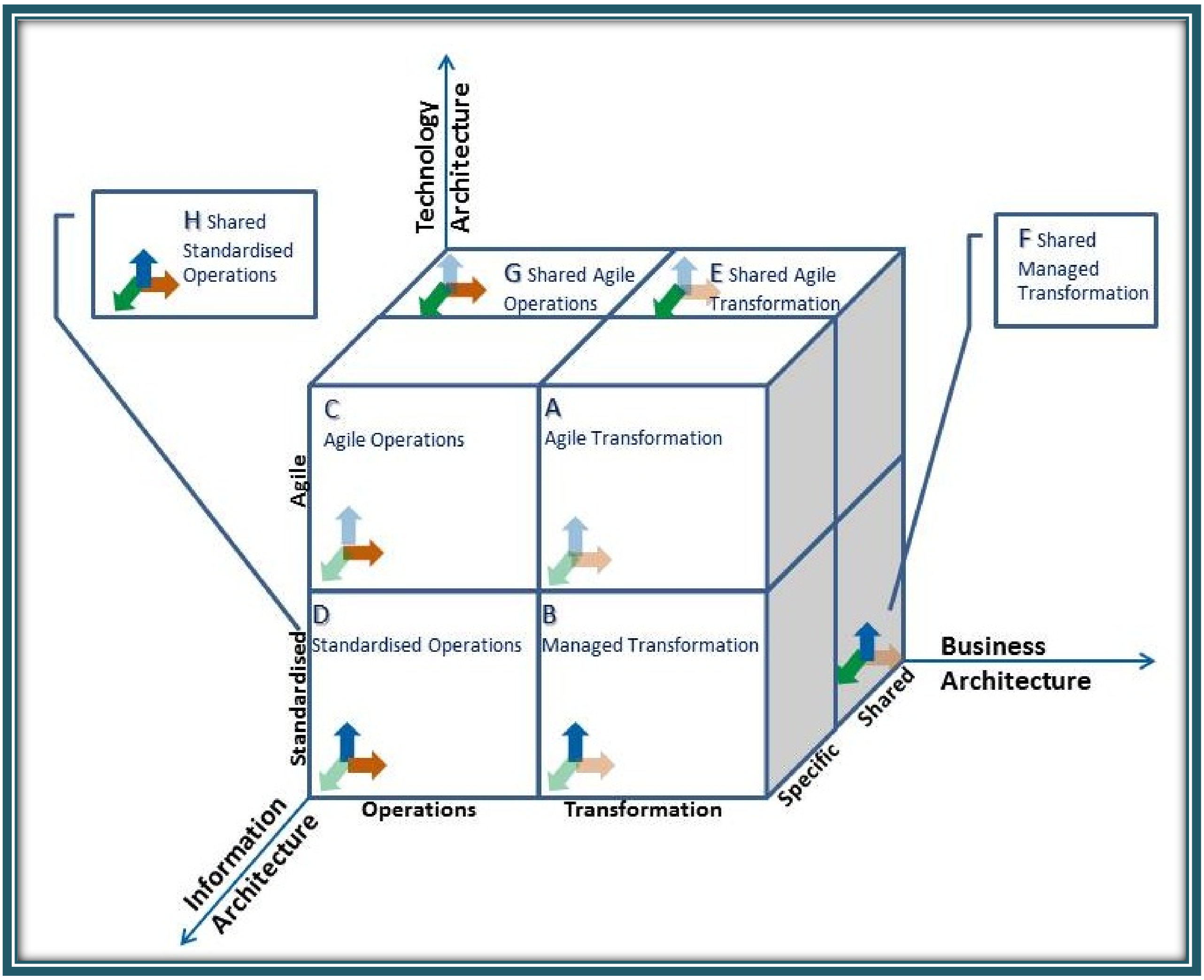


Figure 15-10: Design Alternative C

Third, aspects of the design alternatives were **combined** and **utilised** in order to benefit from the best properties of each design and to form the basis of the proposed Relational Enterprise Architecture Metamodel (REAM) (refer to paragraph 8.2.3).

Fourth, the REAM was **developed** (paragraph 8.2.3). The depiction of the REAM is repeated here as a hexagonal prism net in Figure 8-11 (repeated here as Figure 15-11). The architectural domains business architecture, information architecture, technology architecture and solutions architecture as well as the architecture processes and governance are depicted in different colours. The different shades indicate the level of detail. The current and future states are depicted as vertical bands within each domain. Different views (including viewpoints) are depicted as horizontal bands (annuli) across all the domains. Relations are indicated with chains between adjoining domains.

Figure 15-11: Proposed Relational EA Metamodel

Each area is numbered for easier reference. A summary of the areas are tabled in Table 8-10 (repeated here as Table 15-10).

AREA	DESCRIPTION	AREA	DESCRIPTION
P1	Architecture processes and governance	I11	Future IA of View B
Business Architecture		I12	IA & BA part of View B
B1	Overview of BA and IA relation	I13	IA & TA part of View C
B2	Overview of BA current state	I14	Current IA of View C
B3	Overview of BA future state	I15	Future IA of View C
B4	Overview of BA and TA relation	I16	IA & BA part of View C
B5	BA & IA part of View A	I17	Detailed reference architecture of IA & TA relation
B6	Current BA of View A	I18	Detailed reference architecture of current IA
B7	Future BA of View A	I19	Detailed reference architecture of future IA
B8	BA & TA part of View A	I20	Detailed reference architecture of IA & BA relation
B9	BA & IA part of View B	**Technology Architecture**	
B10	Current BA of View B	T1	Overview of TA and BA relation
B11	Future BA of View B	T2	Overview of TA current state
B12	BA & TA part of View B	T3	Overview of TA future state
B13	BA & IA part of View C	T4	Overview of TA and IA relation
B14	Current BA of View C	T5	TA & BA part of View A
B15	Future BA of View C	T6	Current TA of View A
B16	BA & TA part of View C	T7	Future TA of View A
B17	Detailed reference architecture of BA & IA relation	T8	TA & IA part of View A
B18	Detailed reference architecture of current BA	T9	TA & BA part of View B
B19	Detailed reference architecture of future BA	T10	Current TA of View B
B20	Detailed reference architecture of BA & TA relation	T11	Future TA of View B
Information Architecture		T12	TA & IA part of View B
I1	Overview of IA and TA relation	T13	TA & BA part of View C
I2	Overview of IA current state	T14	Current TA of View C
I3	Overview of IA future state	T15	Future TA of View C
I4	Overview of IA and BA relation	T16	TA & IA part of View C
I5	IA & TA part of View A	T17	Detailed reference architecture of TA & BA relation
I6	Current IA of View A	T18	Detailed reference architecture of current TA
I7	Future IA of View A	T19	Detailed reference architecture of future TA
I8	IA & BA part of View A	T20	Detailed reference architecture of TA & IA relation
I9	IA & TA part of View B		
I10	Current IA of View B	S1	Solution Architecture

Table 15-10: Summary of Areas in the REAM

Lastly, the proposed REAM was **described** in paragraph 8.3, according to the headings below.

- Standards – the IEEE Std 1471-2000 was applied.
- Documentation – all architecture documentation should be structured and available in a repository. A template was provided and every area of the REAM was detailed.
- Architecture Principles – a template for describing architecture principles as well as a categorised list of example principles were provided.
- Definition of Terms – definitions for the relevant terms in the REAM were provided.
- Depictions – the metamodel was depicted as a geographical net and with photographs.
- Viewpoint Specifications – the how and what of viewpoints were discussed and examples were provided.
- Architectural Views – different examples and the application in the REAM were provided.
- Alignment Approach – the REAM's integration strategies and interlinking mechanisms were highlighted.
- Stakeholders and Concerns – stakeholders and their concerns form part of every area of the REAM.
- Applying the metamodel – the use of TOGAF's ADM is proposed as a mechanism for applying the REAM.

The <u>third section</u> entailed the empirical testing of the metamodel, the modification of the REAM and the applicability of the REAM at the University of Pretoria. The research methodology is described in detail in Chapter 9. This section addressed the following sub-research questions:

*k) What is the **status quo** of enterprise architecture in the case study enterprises?*

The three case studies are documented in Chapters 10, 11 and 12 respectively. The EA *status quo* in the three Case Studies is summarised in Table 15-11 below.

ORGANISATION	EA STATUS QUO
Case Study A	Well-established EA function, large architect teams, centralised EA governance, with the exception of BA
Case Study B	Established EA function with three teams and centralised EA governance.
Case Study C	Well-established EA function, large architecture teams and centralised EA governance.

Table 15-11: Summary of EA *Status Quo* in Case Studies

*l) What are the **practical application possibilities** of the proposed integration metamodel within the case study enterprises?*

The perceived application possibilities of the REAM by the three Case Studies are summarised in Table 15-12 below.

ORGANISATION	APPLICATION POSSIBILITIES
Case Study A	A comprehensive abstract presentation of EA, which can be used very effectively within the architecture community or an architecture practice. It can be applied to plot different aspects of EA and measure the completeness, governance and maturity of EA in an enterprise.
Case Study B	Application is dependent on the detail behind the high-level diagram. It could be used as an aggregation model and useful in depicting different views of an enterprise's EA.
Case Study C	Enhancing the understanding and skill level of EA, plotting EA quality, gaps and maturity and forming strong linkages between the architectural domains.

Table 15-12: Summary of Application Possibilities in Case Studies

*m)How was the proposed metamodel **received** in the case study enterprises?*

The responses of the three Case Studies to the REAM are summarised in Table 15-13 below.

ORGANISATION	RESPONSE TO REAM
Case Study A	Received it enthusiastically and concurred that interlinking is a very important, and somewhat neglected, area of EA. The REAM sparked their interest to plot some of their EA aspects, such as governance
Case Study B	Received it as clear to architects and as input to discuss their own EA operations, for example, the hand-over or borders between architectural domains.
Case Study C	Received it enthusiastically and called it clever, innovative and refreshing.

Table 15-13: Summary of Response to the REAM by Case Studies

*n) What are the possible **limitations** of the proposed metamodel within the case study enterprises?*

The perceived possible limitations of the REAM by the three Case Studies are summarised in Table 15-14 below.

ORGANISATION	PERCEIVED LIMITATIONS
Case Study A	Not indicating transitional states explicitly, not covering the whole EA life cycle, not providing a clear methodology and not providing sufficient detail regarding the pink links.
Case Study B	Not indicating transitional states, and having a lack of detail and methodology
Case Study C	The lack of providing specific viewpoints as well as operational context.

Table 15-14: Summary of Perceived Limitations of the REAM by Case Studies

*o) How was the proposed metamodel **modified**, based on the input from the case studies?*

In paragraph 13.2, all the enhancement/modification inputs from the Case Studies were combined in a list according to the categories: enhancements in depiction, clarification, additional features and expansion possibilities. Each input was addressed individually and Table 15-15 provides a summary of the inputs, number of occurrences and actions.

The result of these inputs and actions is a modified REAM (see paragraph 13.4):

- depicted in Figure 13-7 (repeated here as Figure 15-12);
- areas tabled in Table 13-1 (repeated further below in Table 15-16); as well as
- a context diagram depicted in Figure 13-9 (repeated lower down in Figure 15-13).

INPUT	#	*ACTION*
DEPICTION		
Indicate multiple transition states	3	The diagram was changed to indicate the short, medium and long-term future.
Revisit numbering	2	The order of the numbering was changed.
Enhance visibility of interlinking	2	The graphic & legend were changed and links were added.
Provide more detail	2	A limited level of detail was provided to Case Studies as per research methodology.
Indicate dynamic area size	1	A note was added to the legend.
CLARIFICATION		
The term 'coherence'	1	The term was changed to 'View Level'.
The purpose of the metamodel	1	A subtitle was provided.
The excluded dimensions	1	A note was added to the legend.
ADDITIONAL FEATURES		
Provide context diagram	3	A context diagram was created.
Add methodology/process	2	Existing methodologies were proposed, such as ADM.
Add 'affected' and 'affected by' relationships	1	This needs to be added in drill-down views and not on diagram.
Add security to linkages	1	This needs to be added in drill-down views and not on diagram.
Add RACI role indicators	1	RACI can be used to indicate roles for specific implementation.
EXPANSION POSSIBILITIES		
Provide physical/3-D software model	2	This should be addressed in a future project.
Provide specific views/viewpoints	1	The REAM is intended to be flexible.
Provide examples in each area	1	This could be addressed in a future project.
Compare the areas & dimensions with TOGAF	1	This could be considered for a future research project.
Develop a maturity evaluation method	1	This could be considered for a future research project.

Table 15-15: Summary of Suggested Enhancements

Figure 15-12: The REAM v1.0

AREA	DESCRIPTION	AREA	DESCRIPTION
P1	Architecture processes and governance	I15	Medium-term future IA of View *b*
Business Architecture		I16	IA & BA part of View *b* & long-term future
B1	Overview of BA:IA relation & current state	I17 …	IA & TA part of View *c …* & current state
B2	Overview of BA short-term future state	I18 …	Short-term future IA of View *c …*
B3	Overview of BA medium-term future state	I19 …	Medium-term future IA of View *c …*
B4	Overview of BA:TA relation & long-term future	I20 …	IA & BA part of View *c …* & long-term future
B9	BA & IA part of View *a* & current state	I5	Detailed RA of IA:TA relation & current state
B10	Short-term future BA of View *a*	I6	Detailed RA of short-term future-state IA
B11	Medium-term future BA of View *a*	I7	Detailed RA of medium-term future-state IA
B12	BA & TA part of View *a* & long-term future	I8	Detailed RA of IA:BA relation & long-term future
B13	BA & IA part of View *b* & current state	**Technology Architecture**	
B14	Short-term future BA of View *b*	T1	Overview of TA:BA relation & current state
B15	Medium-term future BA of View *b*	T2	Overview of TA short-term future-state
B16	BA & TA part of View *b* & long-term future	T3	Overview of TA medium-term future-state
B17 …	BA & IA part of View *c …* & current state	T4	Overview of TA:IA relation & long-term future
B18 …	Short-term future BA of View *c …*	T9	TA & BA part of View *a* & current state
B19 …	Medium-term future BA of View *c …*	T10	Short-term future TA of View *a*
B20 …	BA & TA part of View *c …* & long-term future	T11	Medium-term future TA of View *a*
B5	Detailed RA of BA:IA relation & current state	T12	TA & IA part of View *a* & long-term future
B6	Detailed RA of short-term future-state BA	T13	TA & BA part of View *b* & current state
B7	Detailed RA of medium-term future-state BA	T14	Short-term future TA of View *b*
B8	Detailed RA of BA:TA relation & long-term future	T15	Medium-term future TA of View *b*
Information Architecture		T16	TA & IA part of View *b* & long-term future
I1	Overview of IA:TA relation & current state	T17 …	TA & BA part of View *c…* & current state
I2	Overview of IA short-term future state	T18 …	Short-term future TA of View *c …*
I3	Overview of IA medium-term future state	T19 …	Medium-term future TA of View *c …*
I4	Overview of IA:BA relation & long-term future	T20 …	TA & IA part of View *c …* & long-term future
I9	IA & TA part of View *a* & current state	T5	Detailed RA of TA:BA relation & current state
I10	Short-term future IA of View *a*	T6	Detailed RA of short-term future-state TA
I11	Medium-term future IA of View *a*	T7	Detailed RA of medium-term future-state TA
I12	IA & BA part of View *a* & long-term future	T8	Detailed RA of TA:IA relation & long-term future
I13	IA & TA part of View *b* & current state		
I14	Short-term future IA of View *b*	S1	Solution Architecture

Table 15-16: Summary of Areas in the Modified REAM

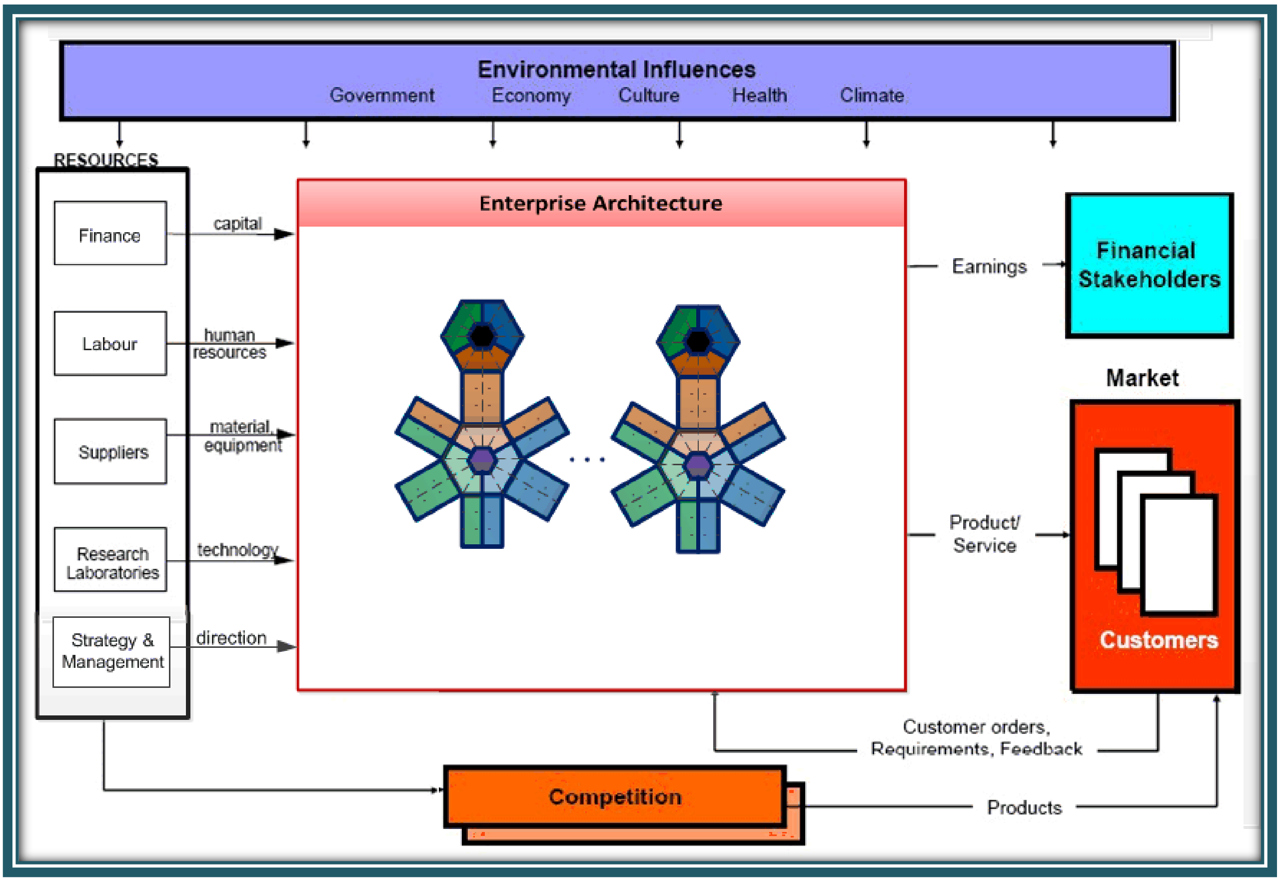


Figure 15-13: REAM(s) in a Context Diagram (adapted from Rummler & Brache, 1995: 10)

*p) What is the applicability of the metamodel to the **University of Pretoria**?*

This sub-research question was addressed by following the same case study methodology with the University of Pretoria (UP) – see Chapter 14. UP views the application possibilities of the REAM to be the creation of structure and order, the flexibility of implementation scope and methodology, the enhancing of interlinking and to use as a tool to communicate about EA. The complete composite diagram might be a bit too complex. Possible enhancement of the REAM could include the numbering of the links and the provisioning of different views of the REAM depending on the audience. The possibilities of achieving a synergy between, for example, the roadmaps or pace layering, and the REAM were discussed – this can be addressed in the future.

*q) What are the **conclusions** of the research?*

The conclusions are discussed below in paragraph 15.2.2.

r) What are the contributions of the research to the body of knowledge?

The value and contribution of the research are discussed below in paragraph 15.3.2.

s) What future research possibilities flow from this research?

Recommendations regarding future research possibilities are addressed below in paragraph 15.4.

15.2.2 Broader Findings

The broader findings are a broader discussion and synthesis relating to the original research question. The above-mentioned findings on the nineteen sub-research questions can now be synthesised into providing a coherent answer to the main research question.

The importance and implications of these findings are:

- EA is a relatively new discipline/sub-discipline with a variety of definitions, frameworks and models.
- The role of EA is to gain a holistic vision of the future state of the enterprise, to capture the complexity of an enterprise and to establish a structured framework for the application of architecture.
- The main functions of EA are to provide alignment, drive innovation, create agility, reduce risk and improve processes, information-provisioning and utilisation of resources.
- The application of EA has a list of benefits, which can be deduced from the main functions. Furthermore documenting EA provides a foundation for execution, provides the capturing of intellectual property and provides metrics.
- There are challenges in documenting an enterprise's EA, such as ambiguity, scalability, the rate of change, maintenance, co-operation and governance.
- Business architecture is an important part of enterprise architecture as it describes the context and environment of the enterprise, which is to be captured by EA.

Business architecture has specific roles within an enterprise and can provide significant benefits to the enterprise. There are different frameworks, models, ontologies and descriptive languages, which can be utilised to describe an enterprise's business architecture and create architecture artefacts.

- Information architecture is an important part of successful enterprise architecture as it describes and governs all the different information sources and systems that are essential for achieving business goals and engagement with the users/clients. Flexibility and agility are important gains from implementing information architecture successfully. Information architecture is included in most of the well-known enterprise architecture frameworks, but there are also specific information architecture frameworks, models, ontologies and descriptive languages for implementing IA.
- Technology architecture has specific roles within an enterprise and can provide significant benefits to the enterprise. There are different frameworks, models, ontologies and descriptive languages, which can be utilised to describe an enterprise's technology architecture and create architecture artefacts. Technology architecture is an important part of enterprise architecture as technology architecture is the realisation of solutions for business requirements via IT infrastructure and systems through IT planning and governing. Technology architecture thus completes the cycle for aligned business requirement fulfilment.
- There is a variety of depictions of EA to illustrate the relationships between the different architectural domains. The proposed single integrated depiction of EA adds value as it provides a holistic view on EA, its main domains and its possible sub-domains in one depiction. This is essential for understanding, promoting and conveying EA within the enterprise and executive. It also attempts to express the interrelations and alignment between the main domains, which aggregate in solutions architecture.
- The terms used to define the relationships between the architectural domains, as well as the role and benefits thereof were discussed and a definition followed.
- The relation between architectural domains has been incorporated in architecture frameworks by making use of integration strategies, such as two and three dimensional matrices, metamodels, viewpoints, service layers and reference and

solution architecture and by making use of interlinking mechanisms, including human intervention, common language, iterative and re-usable processes and blueprints.

- An iterative design process was designed and followed to design a metamodel, including concepts such as research, development, evaluation and refining. The metamodel should include a variety of elements and deliverables in order for the applying enterprise to document their architecture effectively and efficiently.
- The modelling process was followed to create three design alternatives, based *inter alia* on relevant input from existing EA frameworks. These alternatives and the experience gained were utilised to create the proposed Relational Enterprise Architecture Metamodel (REAM). The REAM was described using the following elements: standards, documentation, architecture principles, definition of terms, depictions, viewpoint specifications, views, alignment approach, stakeholders and concerns and mechanisms to apply the metamodel. The REAM provides a multi-level, three-dimensional metamodel including different states, different views, reference architecture, solution architecture, inter-relations and architecture processes and governance.
- The REAM was empirically tested in three Case Studies in three different economic sectors in enterprises with established EA functions.
- The Case Studies produced a list of application possibilities for the REAM, including a metamodel to plot EA, measure maturity, enhance EA governance, depict different views of an enterprise's EA, make gaps visible, form strong linkages and to provide structure and order to EA.
- The Case Studies received the REAM well and understood the concepts easily.
- The Case Studies indicated a few perceived limitations of the REAM, including transition states, lack of methodology and visibility of detail.
- The Case Studies produced a list of possible enhancements or modification to the REAM. These were categorised, evaluated and reacted upon.
- Since all three Case Studies accepted the REAM in principle, the inputs received were used to refine the REAM and not change it substantially. The valuable inputs were utilised to modify and enhance the REAM, without over-complicating it.
- The EA function at the University of Pretoria is relatively immature and the REAM could be applied to create structure and order, to address specific focus areas, to enhance interlinking and to communicate the EA concept.

The main research question is: *What should the key characteristics be of a metamodel for enterprise architecture, which focuses on the interfaces between the different architectural domains?* This question was answered:

- first, by defining the terms enterprise architecture, its different architectural domains, their relationships (interfaces) and general and respective existing frameworks and models;
- second, by designing and developing a new metamodel (the REAM) with the background information as input; and
- third, by empirically testing the metamodel through case studies, refining the metamodel and determining its applicability to the University of Pretoria (UP).

In short, to answer the main research question, a new metamodel (the REAM) was conceptualised, developed, tested and refined to showcase the different dimensions and characteristics of an integrated EA metamodel and to focus on the interlinking between the different architectural domains. It is evident, through the three anonymous Case Studies and the UP Case Study, that the REAM is useful, places emphasis on the interlinking between the architectural domains and is applicable to enterprises with both mature and immature EA functions.

15.3 EVALUATION OF THE RESEARCH

The evaluation of the research should include the evaluation of the methodology, data collection and analysis and valuing the contribution(s) made (Badenhorst, 2008: 207). The methodology will be discussed first, followed by the value and contribution of the research.

15.3.1 Methodology

Three sound research methodologies were followed during the research:

- **Literature study** – a critical literature analysis of recent relevant literature sources. A comprehensive critical literature study was done on a large number of resources published since the year 2000. The literature study was used to provide background, compile summaries and form definitions. These outputs were used as input to the development of a new metamodel.

- **Action research** – the development of a new metamodel. The methodology for developing the metamodel is described in detail in Chapter 7. It consists of an iterative design process, including concepts such as research, development, evaluation and refining. Three design alternatives were created and used as a basis for developing and documenting the Relational Enterprise Architecture Metamodel (REAM).
- **Empirical case studies** – assessing the applicability of the newly developed metamodel with architecture practices in companies. The case studies were done by using semi-structured interviews and were complemented by document analysis. The methodology choices and process are described in detail in Chapter 9. Information gathered during the case studies was recorded according to the questions in the structured interview. The input gathered from the case studies proposed adjustments or modifications to the REAM. Every input was evaluated and addressed and resulted in a modified REAM. Finally, the possible application of the modified REAM at the University of Pretoria was assessed through the case study methodology.

15.3.2 Value and Contribution

The value of and contributions made by this research are listed below:

- A new definition for enterprise architecture was created as a synthesis from the literature.
- New summaries of the roles, the functions, the benefits, the documenting challenges and benefits of enterprise architecture were produced.
- A new definition for business architecture was created as a synthesis from the literature.
- A new summary of frameworks and models applicable to business architecture was produced.
- A new definition for information architecture was created as a synthesis from the literature.

- A new summary of frameworks and models applicable to information architecture was produced.
- A new definition for technology architecture was created as a synthesis from the literature.
- A new summary of frameworks and models applicable to technology architecture was produced.
- A working taxonomy for enterprise architecture was proposed.
- An integrated depiction of enterprise architecture was proposed.
- A scale of the terms, used to indicate the relationship between architectural domains, was created.
- A new definition for relating architectural domains was created.
- A new summary of the integration strategies and interlinking mechanisms for EA frameworks and integration frameworks was produced.
- A relational EA metamodel (REAM) was designed and described.
- Three Case Studies were executed and contributed to the evaluation and validation of the REAM.
- An improved modified REAM was developed.
- A context diagram for the REAM within an enterprise and environment was created.
- The REAM proved to be also potentially useful for a relatively immature EA enterprise, such as the University of Pretoria.

It is evident, from the above list of contributions, that the research provided new summaries and definitions, as well as an empirically tested and refined integrated EA metamodel.

15.4 RECOMMENDATIONS

It is recommended that the TOGAF ADM is used to implement the REAM, because the "TOGAF ADM is the result of continuous contributions from a large number of architecture practitioners" (The Open Group, 2009c: 51) and is widely known and used. It is recommended to start with the area(s) which make the most business sense for the enterprise. The ADM makes use of eight phases with steps in each phase. For example, "the steps within the Technology Architecture phase are as follows:

- Select reference models, viewpoint, and tools
- Develop Baseline Technology Architecture Description
- Develop Target Technology Architecture Description
- Perform gap analysis
- Define roadmap components
- Resolve impacts across the Architecture Landscape
- Conduct formal stakeholder review
- Finalise the Technology Architecture
- Create Architecture Definition Document" (The Open Group, 2009c: 54).

The following recommendations for **further research** are derived from insights and inputs gained from the case studies:

- The development of a physical three-dimensional depiction of the REAM.
- The development of a three-dimensional software model of the REAM with drill-down capabilities.
- The development of examples for each of the areas of the REAM to illustrate functionality.
- The compilation of a comparison between the areas of the REAM and the dimensions found in TOGAF.
- The development of an EA maturity model which uses the REAM as a basis.
- The development of detailed descriptions on different levels of detail for the REAM.
- The expansion and enhancement of the REAM by implementing it at the University of Pretoria.
- The development of synergy between roadmaps and the REAM.

15.5 CONCLUDING REMARKS

The nineteen sub-research questions were addressed by utilising three research methodologies, *i.e.* literature study, action research and empirical case studies. In answer to the main research question a new metamodel (the REAM) was conceptualised, developed, tested and refined to showcase the different dimensions and characteristics of an integrated EA metamodel and to focus on the interlinking between the different architectural domains.

The research used sound methodologies and provided an empirically tested integrated EA metamodel to the body of EA knowledge as well as new summaries and definitions. Recommendations were made for further related research projects and implementation.

ADDENDUM A:8.1 GLOSSARY OF TERMS OF THE REAM

Addendum to Chapter 8

Abstraction: "The technique of providing summarized or generalized descriptions of detailed and complex content." A level of abstraction "can also mean providing a focus for analysis that is concerned with a consistent and common level of detail" (The Open Group, 2009c: 21). An example is the overview, coherence and detail level as described by Lankhorst *et al.* (2005: 163).

Architectural Domain: "The architectural area being considered" (The Open Group, 2009c: 25). In the proposed Relational EA Metamodel, the architectural domains are business architecture, information architecture and technology architecture.

Architectural Framework: "A tool for assisting the production of organization-specific architectures. An architectural framework consists of a technical reference model, a method of architectural development and a list of component standards, specifications, products and their interrelationships that can be used to build up architectures" (Blevins, Spencer & Waskiewicz, 2004: 18).

Architecture Governance: "The practice and orientation by which enterprise architectures and other architectures are managed and controlled at an enterprise-wide level" (The Open Group, 2009c: 25).

Architecture Principles: "A qualitative statement of intent that should be met by the architecture" (The Open Group, 2009c: 26). "Architectural principles are statements that express how your enterprise needs to design and deploy information systems across the enterprise" (Pessi, Magoulas & Hugoson, 2011: 54).

Architecture Vision: "A high-level, aspirational view of the Target Architecture" (The Open Group, 2009c: 26). "A succinct and strategic statement describing the targeted end state for the architecture in five years" (California Information Technology Council. Enterprise Architecture and Standards Committee, 2005: 23).

Artefact: General definition: An "object that is made by a person, such as a tool or a decoration, especially one that is of historical interest" (Cambridge Dictionaries Online, 2011). EA definition: "An architectural work product that describes an architecture from a specific viewpoint" (The Open Group, 2009c: 26). "The relevant documentation, models, diagrams, depictions, and analyses, including a baseline repository and standards and security profiles" (USA. Federal Chief Information Officers Council, 2010: 67).

Business Architecture (BA): Business architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships and change/innovation of the business strategies/objectives, processes/value chains, capabilities, functions/structure and resources (*e.g.* HR & finance) of an enterprise (see paragraph 3.2.2).

Catalogue: General definition: "a complete list of items, typically one in alphabetical or other systematic order" (Oxford Dictionaries, 2011). EA definition: "A structured list of architectural outputs of a similar kind, used for reference" (The Open Group, 2009c: 708).

Context: General definition: "the whole situation, background, or environment relevant to a particular event, personality, creation, etc." (Webster's New World Dictionary, 2011). EA definition: "The environment, or *context*, determines the setting and circumstances of developmental, operational, political, and other influences upon that system" (IEEE, 2000: 4).

Deliverable: "A deliverable is a work product that is contractually specified and in turn formally reviewed, agreed, and signed off by the stakeholders" (The Open Group, 2009c: 361). It represents the output.

Enterprise Architecture (EA): Enterprise architecture is the process of describing, modelling, communicating, applying and governing the strategies, processes, current state, future-state blueprints, interrelationships, change/innovation and alignment/integration of the business, information, technology and information systems of an enterprise (see paragraph 2.2.2).

Hexagon: "a plane figure with six straight sides and angles" (Oxford Dictionaries, 2011).

Information Architecture (IA): Information architecture is the process of describing, modelling, communicating, applying and governing the current state, future-state blueprints, interrelationships, change/innovation, usability and sharing of the information assets/content, information activities, and the information audience of an enterprise (see paragraph 4.2.2).

Matrix: General definition: "a grid-like arrangement of elements" (Oxford Dictionaries, 2011). EA definition: "A format for showing the relationship between two (or more) architectural elements in a grid format" (The Open Group, 2009c: 714).

Metadata: "Data about data, of any sort in any media, that describes the characteristics of an entity" (The Open Group, 2009c: 32). "In general, metadata is best understood as 'any statement about an information resource', regardless of what it is being used for, which metadata vocabulary is being used, and how the metadata is represented" (Garshol, 2004: 379).

Metamodel: "A model that describes how and with what the architecture will be described in a structured way" (The Open Group, 2009c: 32). A metamodel "is a subset of constructs that can be mapped to multiple technologies" (Hyam, 2006: 10).

Model: "A model provides a smaller scale, simplified, and/or abstract representation of the subject matter" (The Open Group, 2009c: 33). A model is "a formal specification of the function, structure and/or behavior of an application or system. A model is often presented as a combination of drawings and text. The text may be in a modeling language or in a natural language" (Blevins, Spencer & Waskiewicz, 2004: 19).

Prism: "a solid geometric figure whose two ends are similar, equal, and parallel rectilinear figures, and whose sides are parallelograms" (Oxford Dictionaries, 2011).

Process: General definition: "a series of actions or steps taken in order to achieve a particular end" (Oxford Dictionaries, 2011). EA definition: "a sequence of activities that together achieve a specified outcome, can be decomposed into sub-processes, and can show operation of a function or service" (The Open Group, 2009c: 717).

Repository: General definition: "A place where things are stored and can be found" (Cambridge Dictionaries Online, 2011). EA definition: "A system that manages all of the data of an enterprise, including data and process models and other enterprise information" (The Open Group, 2009a: 33).

Roadmap: General definition: "a plan or strategy intended to achieve a particular goal" (Webster's New World Dictionary, 2011). EA definition: "An abstracted plan for business or technology change, typically operating across multiple disciplines over multiple years" (The Open Group, 2009c: 35).

Solution Architecture: "A solution architecture is an architectural description of a specific solution. SAs combine guidance from different EA viewpoints (business, information and technical)" (Guevara & Robertson, 2011: 4). "A description of a discrete and focused business operation or activity and how IS/IT supports that operation". It typically applies to a single project (The Open Group, 2009c: 37).

Stakeholder: General definition: "A stakeholder is a person who has an interest in or investment in something and who is impacted by and cares about how it turns out" (Webster's New World Dictionary, 2011). EA definition: "An individual, team, or organization (or classes thereof) with interests in, or concerns relative to, the outcome of the architecture" (The Open Group, 2009c: 38).

Technology Architecture (TA): Technology architecture is the process of describing, modelling, communicating, applying, governing and maintaining the current state, future-state blueprints, interrelationships and change/innovation of the IT systems, infrastructure, strategy, portfolio and investment of an enterprise (see paragraph 5.2.2).

View: "The representation of a related set of concerns. A View is what is seen from a viewpoint" (The Open Group, 2009c: 39). "A representation of a whole system from the perspective of a related set of concerns" (IEEE, 2000: 3). For example, executive perspective, business management perspective, architect perspective, technician perspective, *etc.* (Zachman, 2011).

Viewpoint: "A definition of the perspective from which a view is taken" (The Open Group, 2009c: 40). The IEEE (2000: 4) describes a viewpoint as: "A specification of the conventions for constructing and using a view" and "A pattern or template from which to develop individual views by establishing the purposes and audience for a view and the techniques for its creation and analysis".

ADDENDUM A:9.1 SAMPLE INTRODUCTORY LETTER

Addendum to Chapter 9

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Information Technology Services

[date]

Attention: [name]

RE: Permission for researching Enterprise Architecture (EA) Model

I am currently working in the Information Technology Services at the University of Pretoria and am busy with my PhD (IT) studies. I am working on a model to enhance the integration and linking between the domains of business architecture, information/application architecture and technology architecture.

I am visiting a few organisations to test my proposed model in theory in order to improve the model. I would like to request permission to involve your organisation in the research. Your organisation will stay totally anonymous, as will the possible individuals assisting in the research. All information will be handled confidentially. I envisage the following steps and time required:

- Initial contact session between myself and you (2 hours) to brief you and for me to gain an understanding of how your organisation approaches and implements the different components of EA.
- I will take the information thus gathered and apply my proposed model onto it (this may include follow-up correspondence with you to clarify uncertainties or another session, if need be).
- Second contact session with a few selected relevant individuals (2½ hours), where I will provide background, present my proposed model and lead a discussion on the applicability of my model.

- I will modify the proposed model to incorporate new input received and provide feedback to you to confirm if the outcome is documented correctly.

Permission is requested to record the interview sessions on an audio recorder.

Possible benefits to your organisation include exposure to other ways of doing enterprise architecture, collaboration opportunities, adding to the body of EA knowledge and assisting research.

Your willingness and ability to assist in the research will be greatly appreciated.

Yours sincerely,

Mrs Yzelle Roets
Technology Architect
University of Pretoria

Supported by the study leaders:

Prof TJD Bothma | Dr JA Pretorius

Natural Sciences II, Room 5-15
University of Pretoria
PRETORIA 0002
Republic of South Africa

Tel: Number 012 420 5911
Fax: Number 012 420 2041

Email address Yzelle.roets@up.ac.za
www.up.ac.za

ADDENDUM A:9.2 BACKGROUND SUMMARY

Addendum to Chapter 9

Broadly, the research consists of three parts:

- The first part of the research (critical non-empirical study of literature) contains the history of EA and the definitions, role, functions, existing frameworks, models and ontologies of each of enterprise architecture, business architecture (BA), information architecture (IA) and technology architecture (TA). The depictions of and relationships between the different layers/domains of EA were also investigated. The outcome is among others a set of newly compiled definitions, of which I have provided here the definition of EA in Figure A:9.2-1, the relating of EA domains in Figure A:9.2-2 and an ontology in Figure A:9.2-3. This terminology will be adhered to in the case study process.

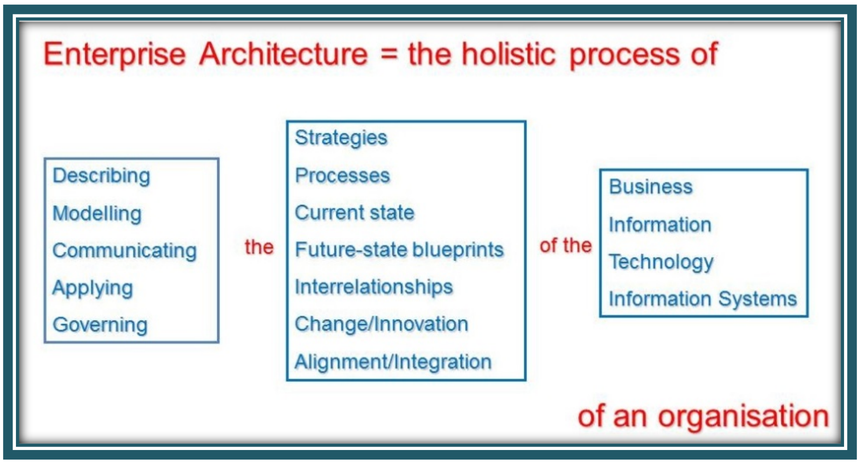


Figure A:9.2-1: Definition of Enterprise Architecture

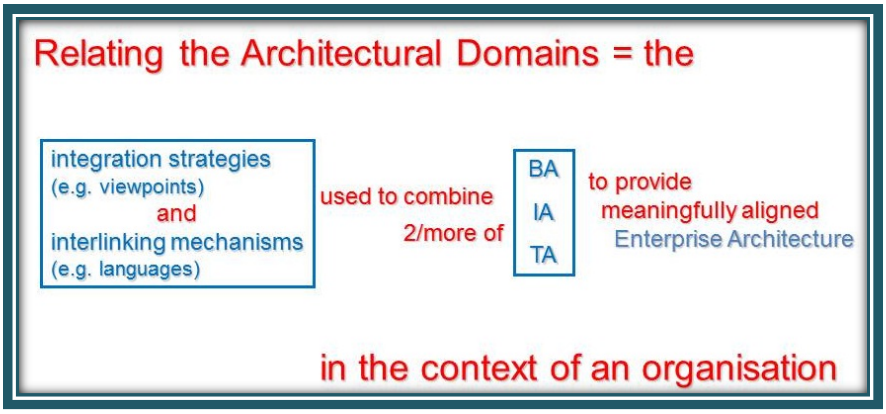


Figure A:9.2-2: Definition Relating the Architectural Domains

Figure A:9.2-3: Sample Ontology

- The second part of the research (model building) contains the design methodology and process, the design of a metamodel, the description and depiction of the metamodel, namely the Relational Enterprise Architecture Metamodel (REAM). The focus of the REAM is on strengthening the interlinking and integration between the different architectural domains.
- The third part of the research (empirical case studies) comprises the methodology and design of the research, the execution of the case studies, the cross-case analyses and the enhanced proposed model.

The purpose of the case studies is to determine the **practical application possibilities** of the proposed integration model within organisations, the possible **shortcomings** of the proposed model, and the **modifications** necessary.

The process will entail the steps as provided in the introductory letter, *i.e.* first contact session, data gathering and analysis, conceptual application of the REAM within the organisation, the second contact session, the modification of the applied REAM and the sign-off.

The second contact session:

- *Purpose*: Obtain feedback from organisation on the applicability of the applied REAM
- *Participants*: A small group of selected relevant individuals from the organisation and the researcher
- *Agenda items*: Introduction of the research project, presentation of the applied REAM and a discussion based on semi-structured questions
- *Duration*: 2½ hours
- *Location:* A venue with a data projector within your organisation, if possible.

ADDENDUM A:9.3 FIRST SESSION INTERVIEW QUESTIONS

Addendum to Chapter 9

Q1 Please explain the enterprise architecture (EA) structure/set-up within your organisation with reference to business architecture, information architecture and technology architecture.
(*e.g.* reporting structures, distinguishing between architectural domains, authority, size of operation)

Q2a Do you make use of a recognised EA Framework/model/methodology?
Q2b If yes, how does this benefit your organisation, and
Q2b how comprehensive is your implementation thereof?
(*e.g.* TOGAF/hybrid/home-grown, training, completeness of implementation)

Q3a How do you govern the EA function within your organisation?
(*e.g.* board, terms of reference, budget allocation, architecture process)
Q3b For example, do you have a formal exception process and what does it entail?

Q4a Do you have a set of architectural principles that has been agreed upon?
Q4b If yes, how and for what purpose do you use them?
(*e.g.* permission to use/copy/see, what do you use them for, how are they enforced/applied, revision cycle, authoritativeness)

Q5a How is EA involved in your project management process?
(*e.g.* not at all, PSA, stage gates, solution sign-off, benefits experienced)
Q5b Do you make use of best practices/reference architectures or something similar?
(*e.g.* standard set of architecture, how you ensure compliance)

Q6a Does EA form part of other organisational and/or decision-making processes?
Q6b If yes, which processes are they?
(*e.g.* technology investigations, strategic planning, budget, capacity planning)

Q7 What mechanisms do you use to ensure consistent integration/interlinking between the different architectural domains?
(*e.g.* none, cross-reference, EARB, co-signoffs, representation)

Q8a Is it necessary for you to distinguishing between different views/viewpoints of EA to derive business value?
Q8b If so, what are they?
(*e.g.* focusses to be applied in REAM, dependent on framework used, management *versus* implementer)

Q9 How do you address, for instance, security in your EA framework/model?
(*e.g.* don't, distributed/specific area, does it reflect the structure of the IT unit)

Q10 How do you determine the architecture for the design and implementation of a new solution?
(*e.g.* solutions architecture, combination of architectural domains for a specific solution, how do you ensure compliance of a new solution)

Q11a Do you have architecture artefacts such as depiction(s) or document(s) which explain your organisation's EA?
Q11b If yes, are you making use of any EA repository/application/tool?
Q11c If so, what are they?
(*e.g.* permission to use/copy/see, visibility within organisation, database, accessibility, indexing)

Q12a Do you document both the current and the future state of architecture?
Q12b If so, where does the focus/emphasis lies?
(*e.g.* percentage of effort, where lies focus, why)

ADDENDUM A:9.4 SAMPLE TEMPLATE FOR APPLIED REAM

Addendum to Chapter 9

AREA	DESCRIPTION	OU/PROCESS	ARTEFACTS
P1	Architecture processes and governance		
Domains	Business Architecture		
	Information Architecture		
	Technology Architecture		
States	Current State		
	Future State		
Views	View A		
	View B		
	View C		
	Interlinking		
	Solution Architecture		

Table A:9.4-1: Sample Template for the Applied REAM

ADDENDUM A:9.5 SECOND SESSION INTRODUCTION

Addendum to Chapter 9

I, Yzelle Roets, am currently working in Information Technology Services at the University of Pretoria and am busy with my PhD (IT) studies (see Figure A:9.5-1 for respective introduction slides). I am working on a model to enhance the integration and linking between the domains of business architecture, information/application architecture and technology architecture. I am visiting a few organisations in order to test my proposed model in theory in order for me to improve the model. I have had a session with [name] to gain an overview of your architecture environment. I have attempted to apply it conceptually to my proposed model, called the REAM. The purpose of this session is to obtain your input and expert opinion on the **practical application possibilities** of the proposed model, the possible **shortcomings** of the proposed model, and the **modifications** necessary to improve it. The intent is thus not to change, evaluate or criticise your way of doing EA (see Figure A:9.5-2 for background slide).

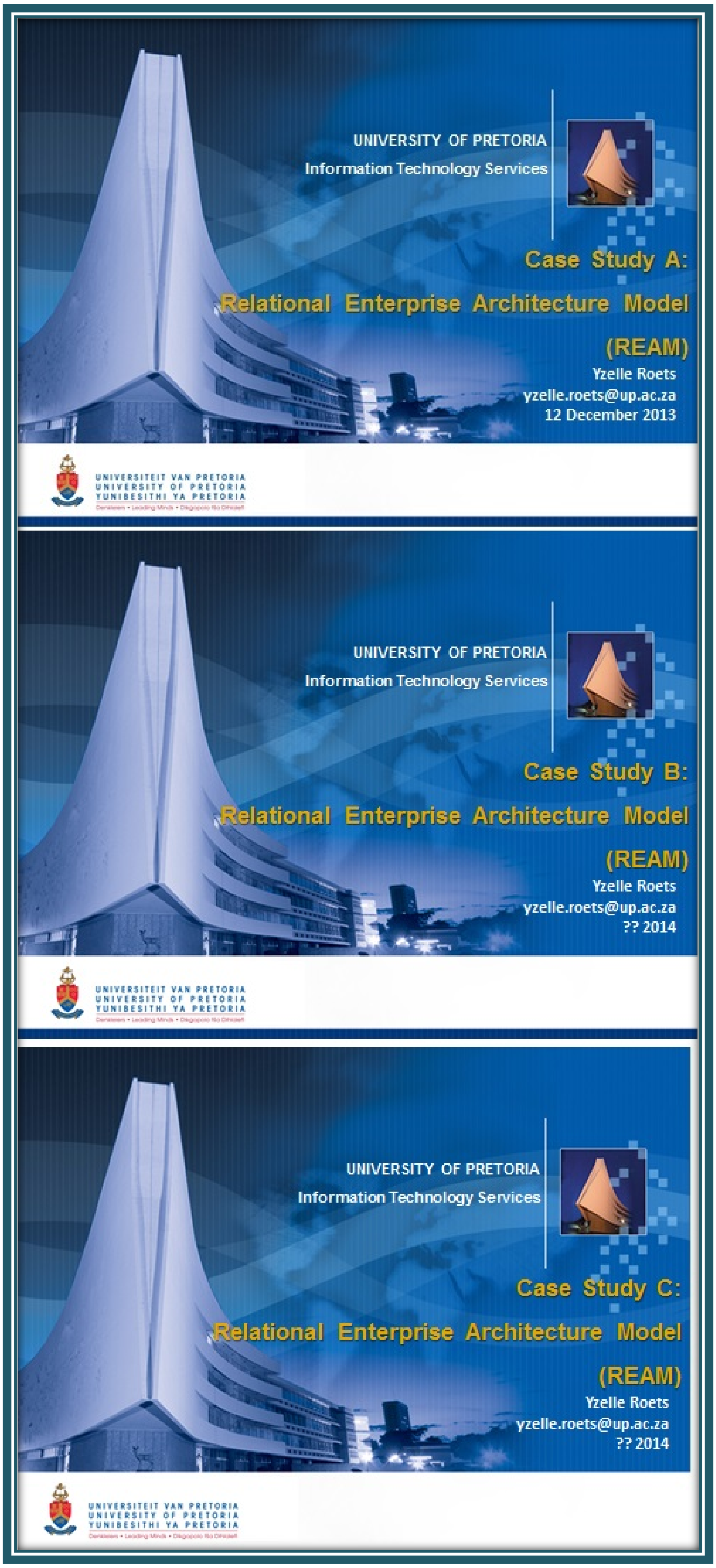


Figure A:9.5-1: Respective Introduction Slides

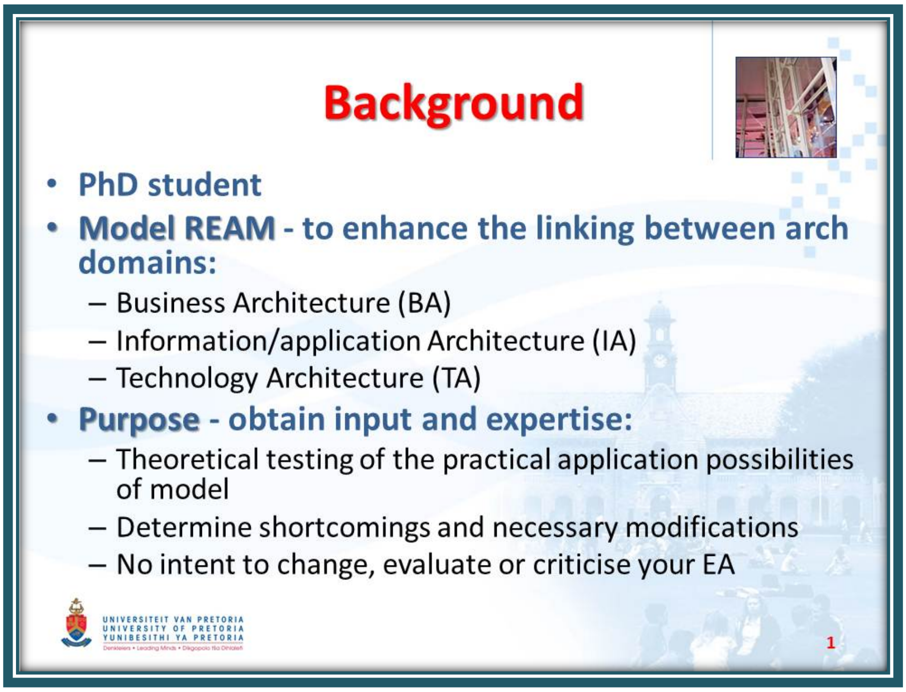


Figure A:9.5-2: Slide 1 – Background

First, I will explain the basic model

- Slide 2 – The REAM is based on a three-dimensional hexagon drawn as a geometrical net to allow better visibility (see Figure A:9.5-3).
- Slide 3 – On the first dimension, the architectural domains are displayed (see Figure A:9.5-4).
- Slide 4 – The second dimension addresses the current and future-state architectures (see Figure A:9.5-5).
- Slide 5 – The third dimension enables the capability to address different architectural views (see Figure A:9.5-6).
- Slide 6 – The last dimension adds the relations (Figure A:9.5-7).
- Slide 7 – Each area is numbered for easier reference (Figure A:9.5-8).

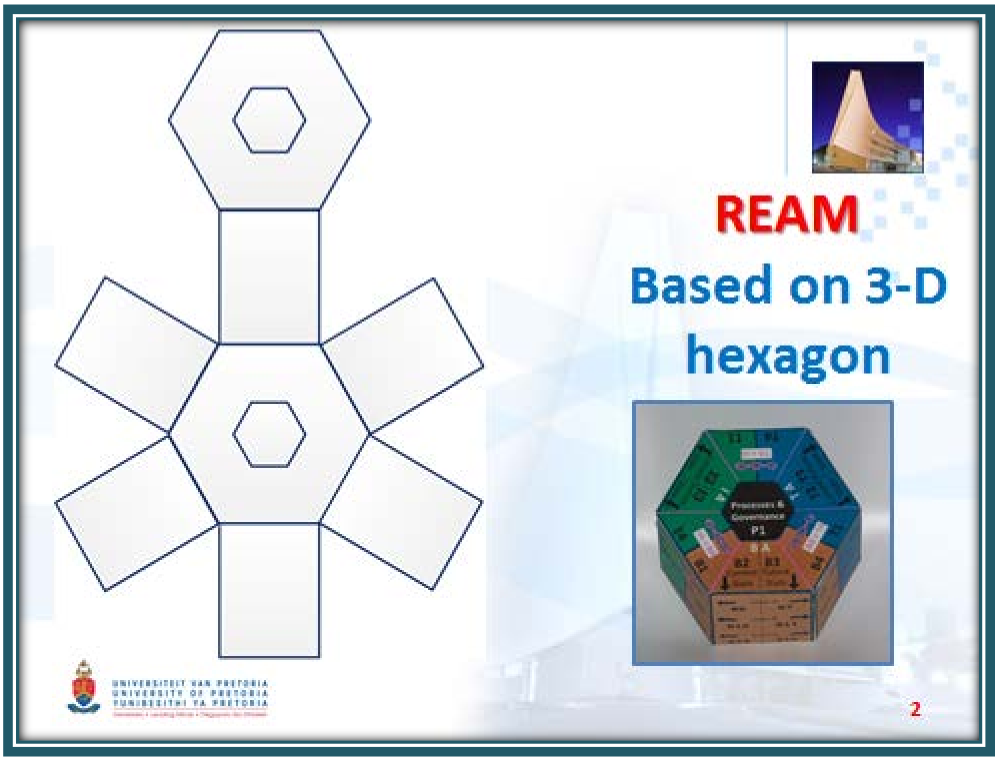


Figure A:9.5-3: Slide 2 – REAM

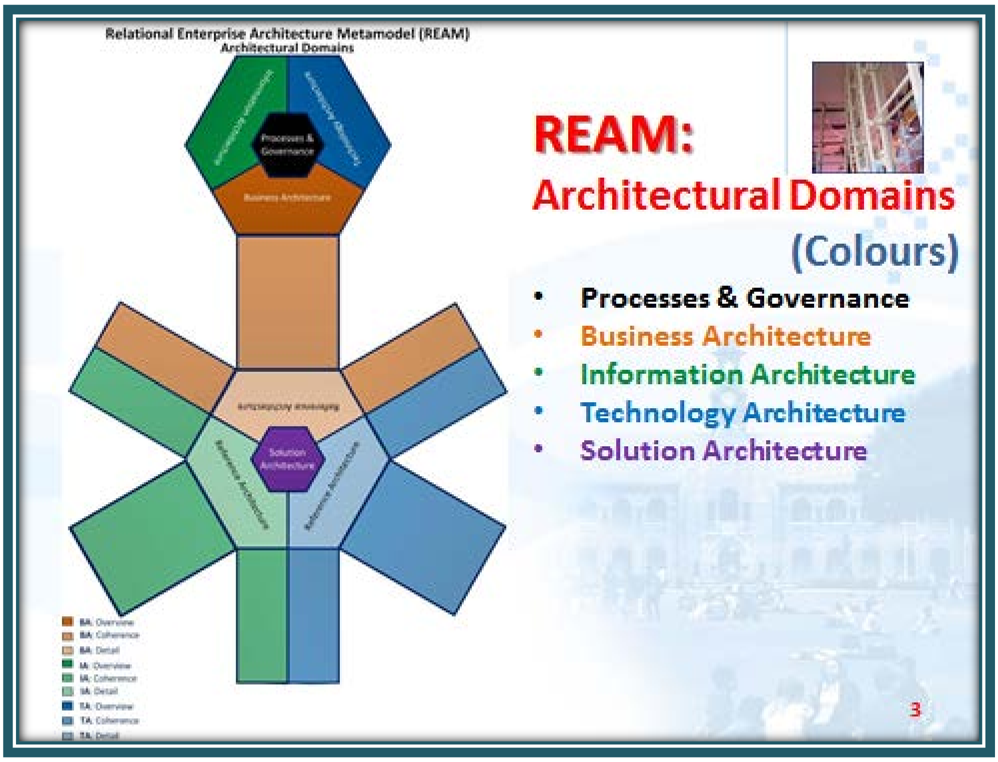


Figure A:9.5-4: Slide 3 – REAM

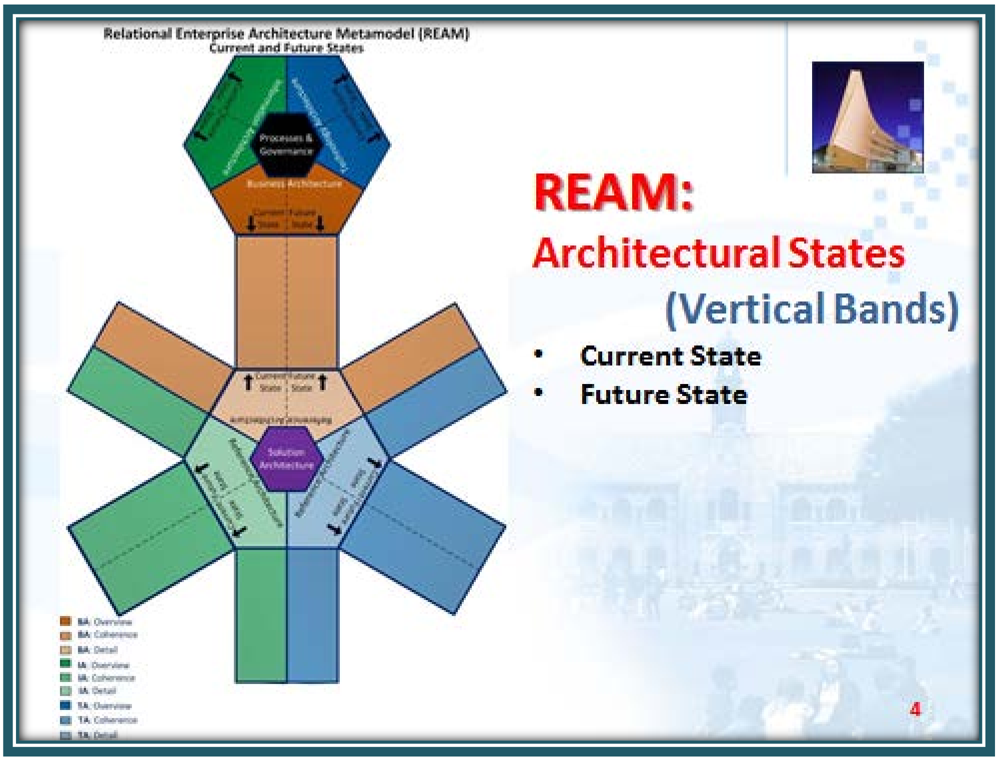


Figure A:9.5-5: Slide 4 – REAM

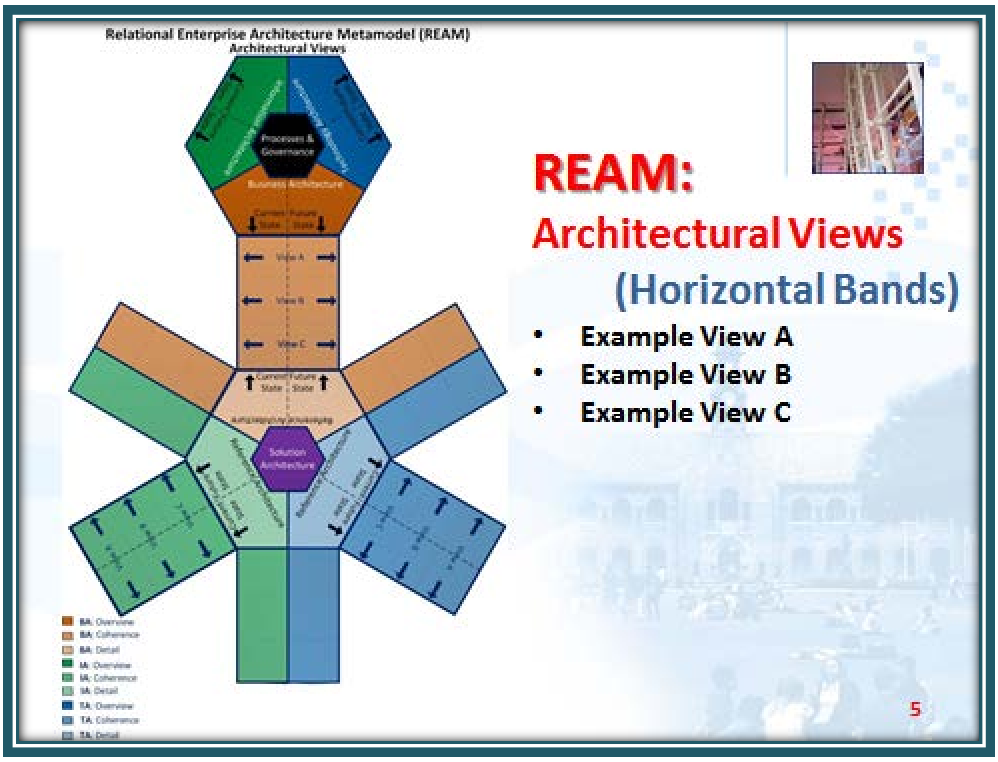


Figure A:9.5-6: Slide 5 – REAM

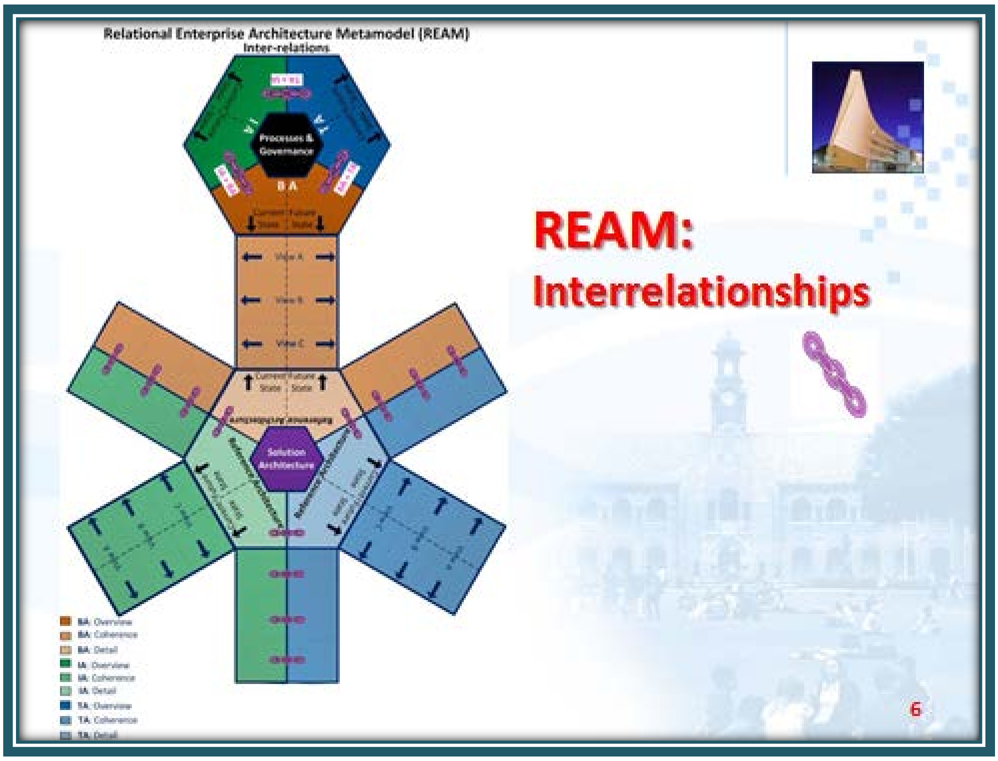


Figure A:9.5-7: Slide 6 – REAM

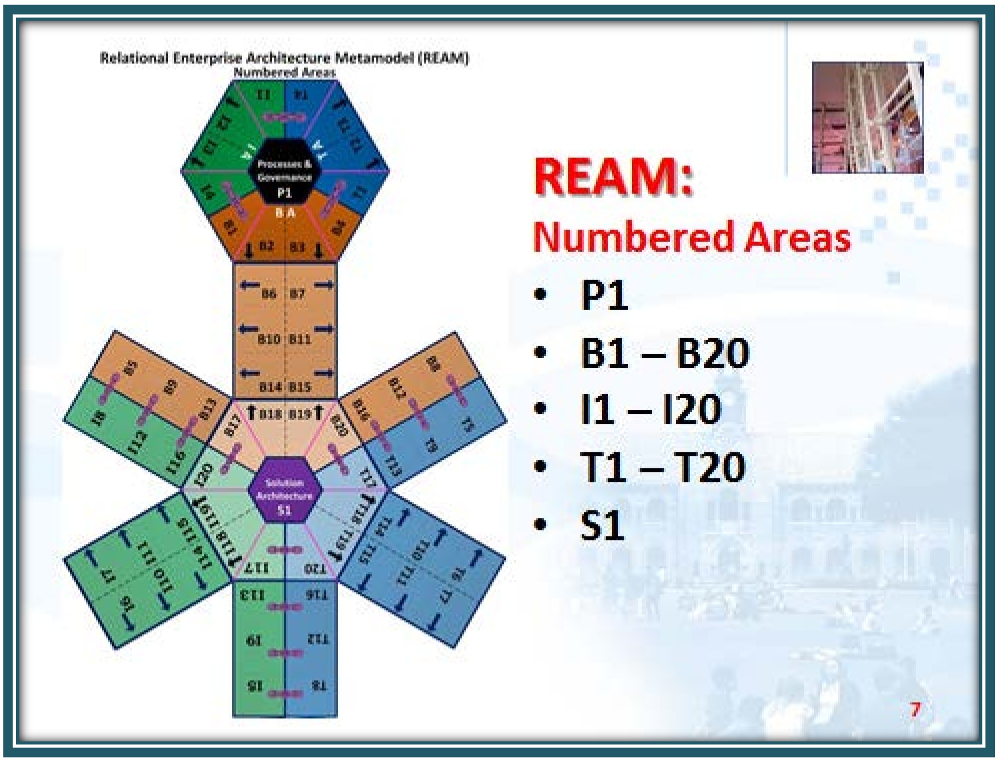


Figure A:9.5-8: Slide 7 – REAM

Second, I have drawn up a table of the areas of the REAM and have plotted your EA environment onto the table in an attempt to illustrate the possible practical value of the

REAM. The outlines of the tables are illustrated below. The detail of each case study is provided within the chapter dealing with that Case Study.

- Slide 8 – P1 Processes & Governance (see Figure A:9.5-9).
- Slide 9 – Architectural domains, *i.e.* Business Architecture, Information Architecture and Technology Architecture (see Figure A:9.5-10).
- Slide 10 – Architectural States (see Figure A:9.5-10).
- Slide 11 – Architectural Views (see Figure A:9.5-12).
- Slide 12 – Interlinking Mechanisms (see Figure A:9.5-13).
- Slide 13 – Solution Architecture (see Figure A:9.5-14).

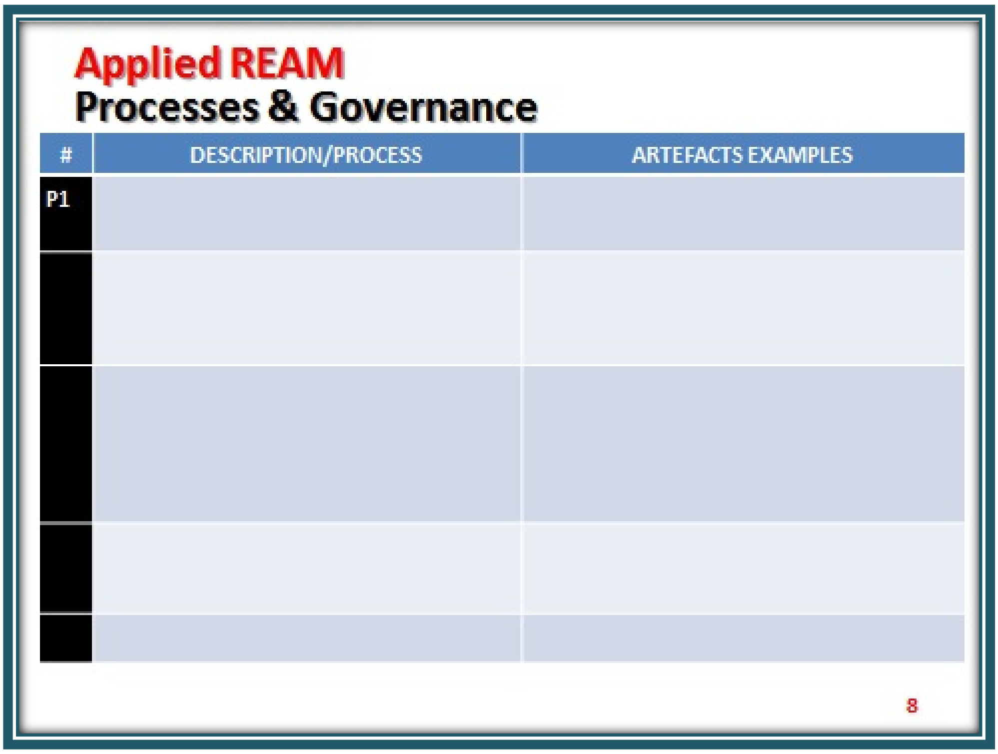


Figure A:9.5-9: Slide 8 – Applied REAM

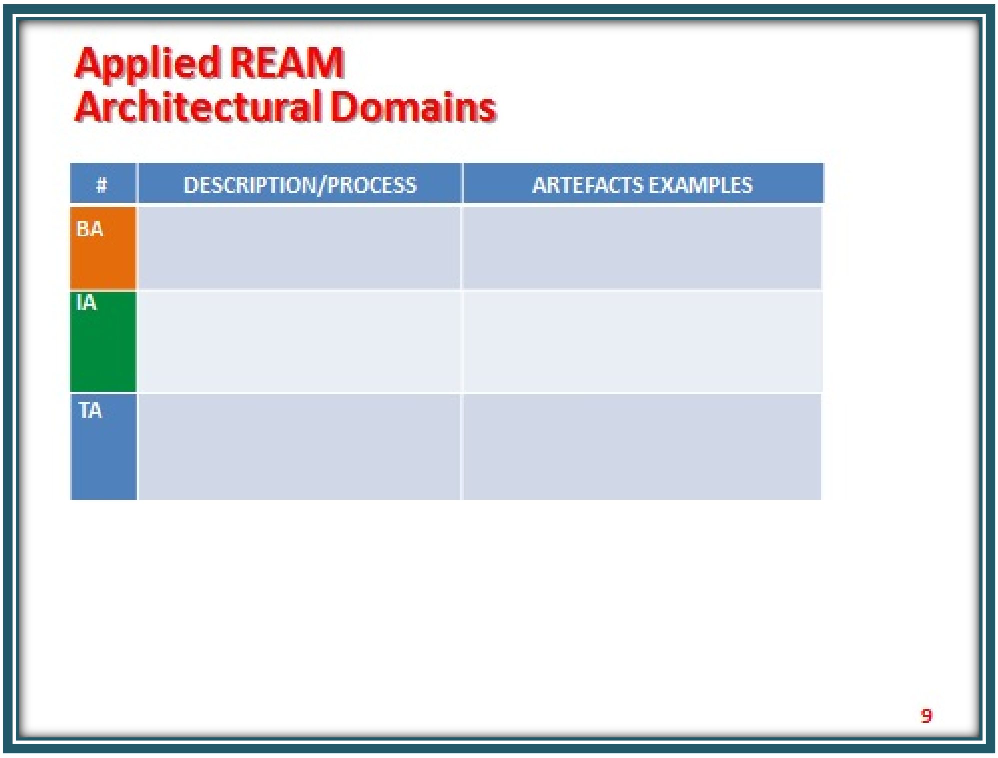


Figure A:9.5-10: Slide 9 – Applied REAM

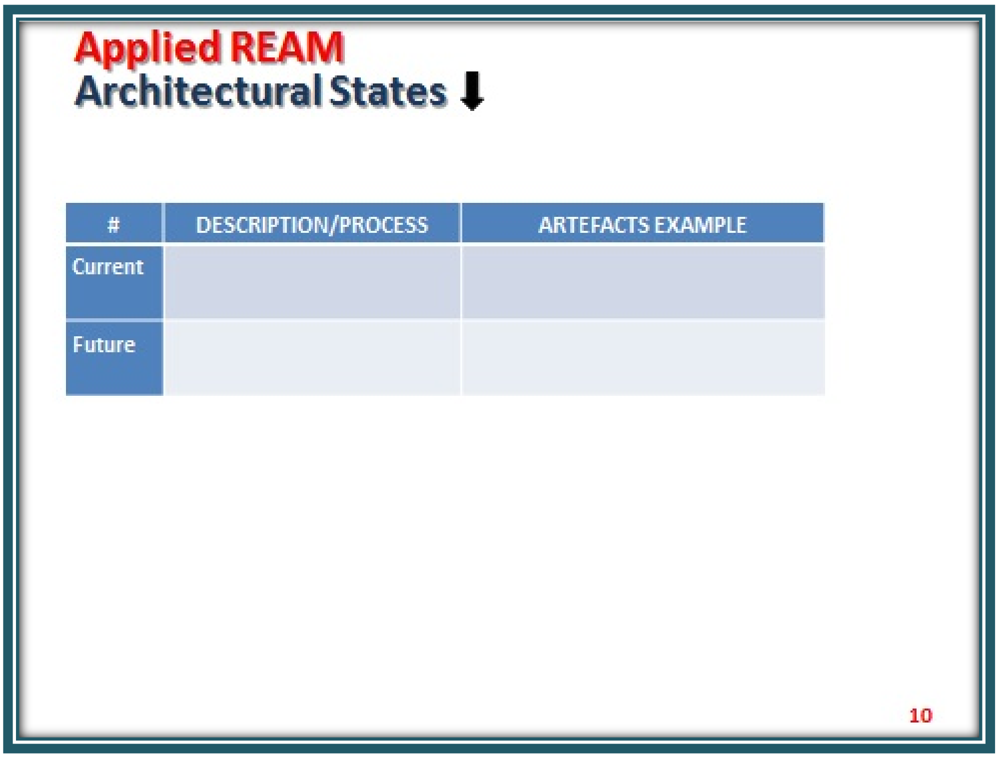


Figure A:9.5-11: Slide 10 – Applied REAM

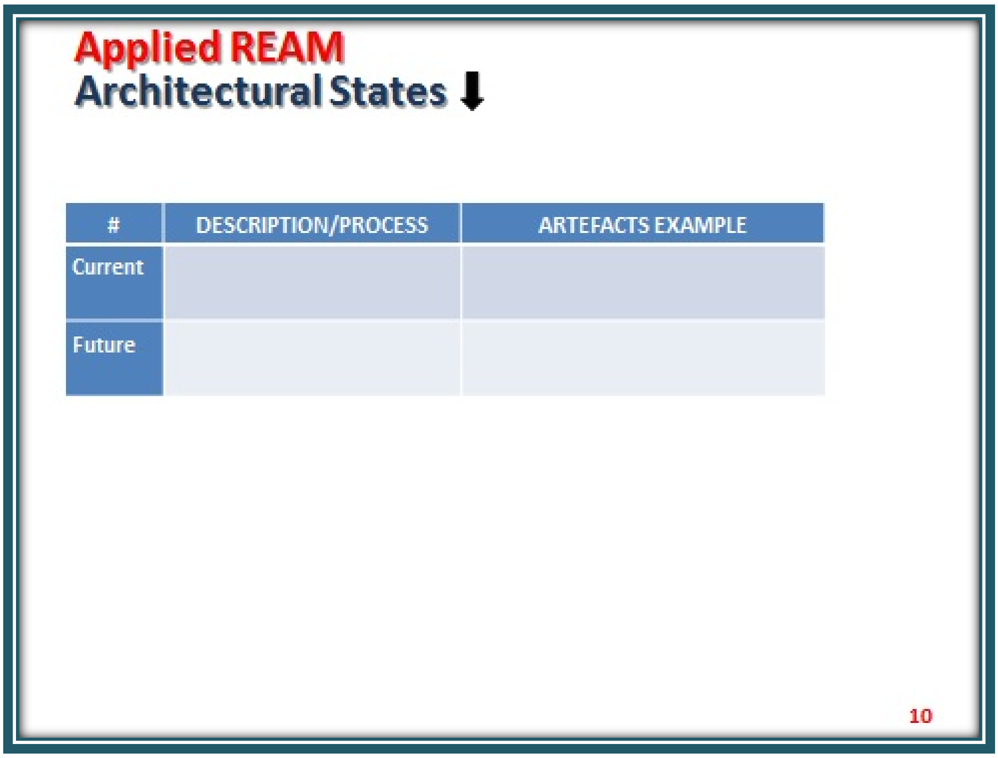


Figure A:9.5-12: Slide 11 – Applied REAM

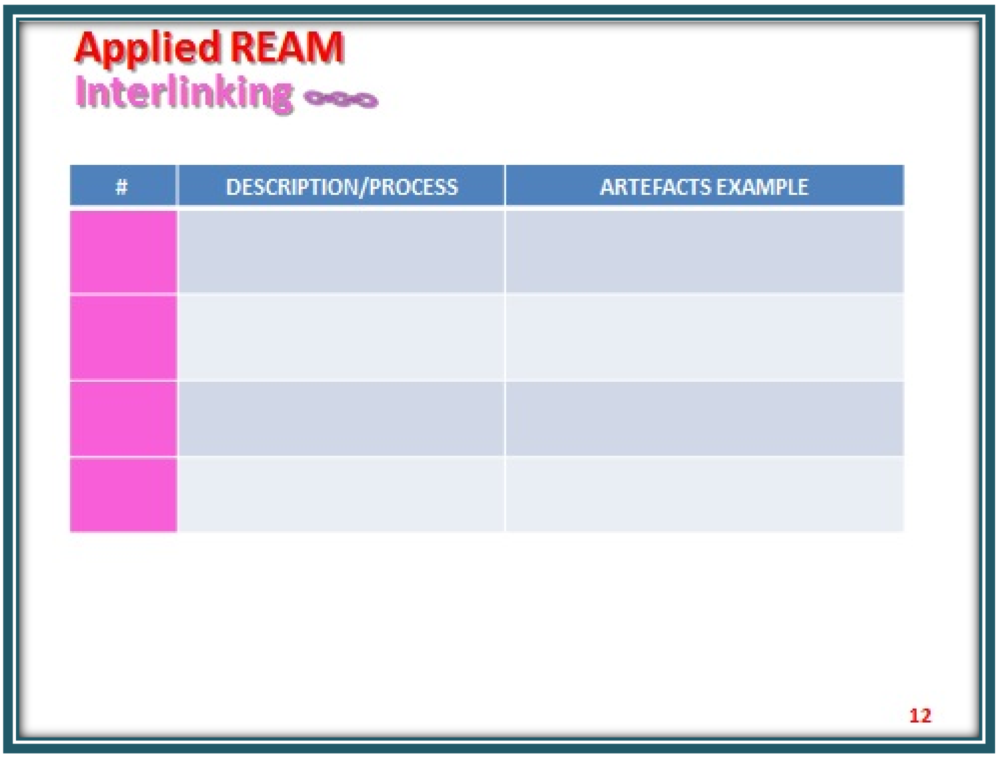


Figure A:9.5-13: Slide 12 – Applied REAM

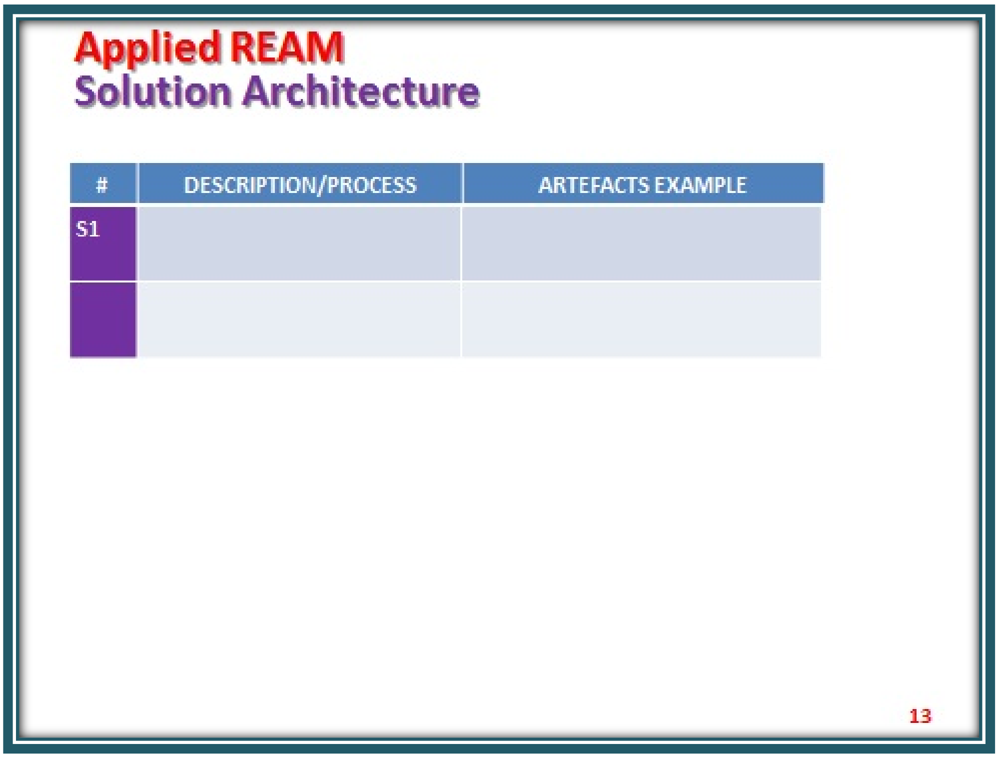


Figure A:9.5-14: Slide 13 – Applied REAM

ADDENDUM A:9.6 SECOND SESSION INTERVIEW QUESTIONS

Addendum to Chapter 9

Q1a Please comment on the potential applicability of the REAM.
(*e.g.* possible applications)

Q1b What challenges do you foresee in the operational application of the REAM?
(*e.g.* overlapping of areas)

Q2 Please comment on the comprehensiveness of the REAM (covering all aspects of EA).
(*e.g.* gaps, enhancements)

Q3 Please comment on the ease of understanding and the ease of use of the REAM.
(*e.g.* too complicated, unclear)

Q4a The REAM provides explicit focus on interlinking. Does this provide value to the EA process?

Q4b Does the application of the REAM improve the mutual influence between the architectural domains?
(*e.g.* no improvement, sensitise to importance of interlinking, creation of artefacts or parts of artefacts describing interlinking, enhance alignment)

Q5a Could the metamodel contribute to improving the governance of EA?

Q5b If yes, how could the metamodel facilitate the governance of EA?
(*e.g.* provides overview, indicates relation, allocation of responsibility for each area)

Q6 What modifications to the REAM would you propose to improve EA in your organisation?
(*e.g.* rename areas, remove areas, add areas, *etc.*)

ADDENDUM A:9.7 SAMPLE SIGN-OFF REQUEST LETTER

Addendum to Chapter 9

Information Technology Services

[date]

[name]

RE: Case Study Enterprise Architecture Metamodel

I want to thank you for your and your group's inputs, contributions and time towards my studies. It was enlightening to have been able to work with you.

I have attached a summary of the first discussion of [date] and a summary of the group discussion of [date]. Could you please confirm in writing that it is a true reflection of the discussions by completing the sign-off below and returning the document to me?

Yours sincerely,

Mrs Yzelle Roets
Technology Architect
University of Pretoria

Signed: Date:

Witness: Date:

Natural Sciences II, Room 5-15
University of Pretoria
PRETORIA 0002
Republic of South Africa

Tel: Number 012 420 5911
Fax: Number 012 420 2041

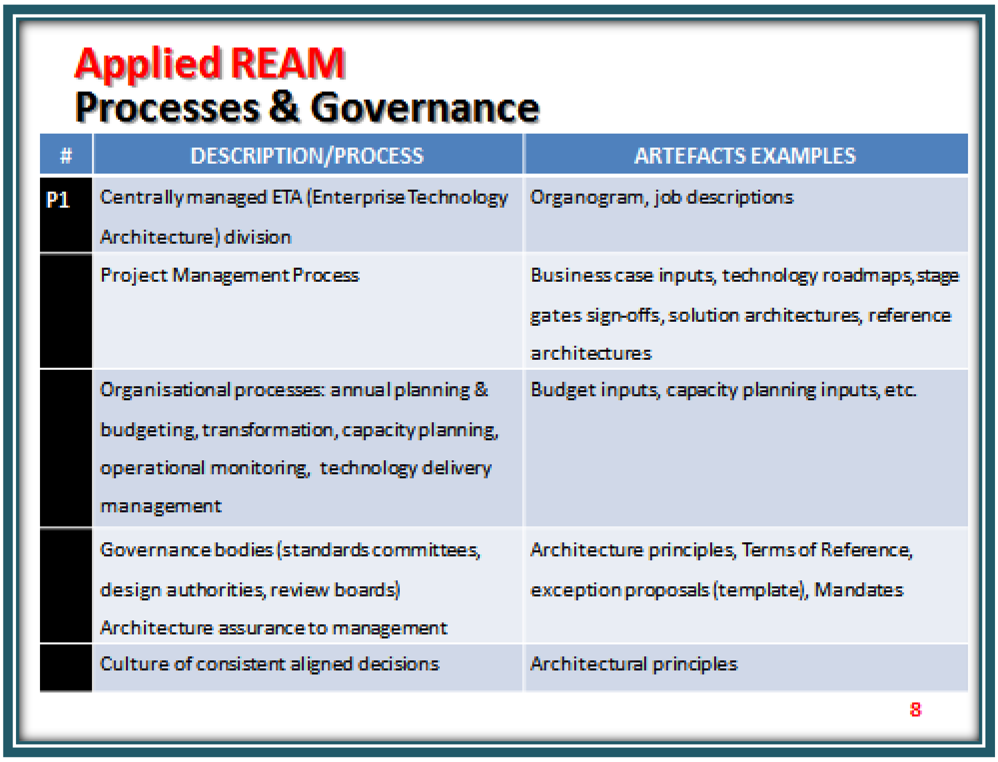
Email address Yzelle.roets@up.ac.za
www.up.ac.za

ADDENDUM A:10.1 SECOND SESSION – APPLICATION OF THE REAM

Addendum to Chapter 10

I have drawn up a table of the areas of the REAM and have plotted my understanding of your EA environment onto the table in an attempt to illustrate the possible practical value of the REAM.

- Slide 8 – P1 Processes & Governance (see Figure A:10.1-1).
- Slide 9 – Architectural domains, *i.e.* Business Architecture, Information Architecture and Technology Architecture (see Figure A:10.1-2).
- Slide 10 – Architectural States (see Figure A:10.1-3).
- Slide 11 – Architectural Views (see Figure A:10.1-4).
- Slide 12 – Interlinking Mechanisms (see Figure A:10.1-5).
- Slide 13 – Solution Architecture (see Figure A:10.1-6).


Applied REAM
Processes & Governance

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLES
P1	Centrally managed ETA (Enterprise Technology Architecture) division	Organogram, job descriptions
	Project Management Process	Business case inputs, technology roadmaps, stage gates sign-offs, solution architectures, reference architectures
	Organisational processes: annual planning & budgeting, transformation, capacity planning, operational monitoring, technology delivery management	Budget inputs, capacity planning inputs, etc.
	Governance bodies (standards committees, design authorities, review boards) Architecture assurance to management	Architecture principles, Terms of Reference, exception proposals (template), Mandates
	Culture of consistent aligned decisions	Architectural principles

8

Figure A:10.1-1: Slide 8 – Case Study A Applied REAM: Processes & Governance

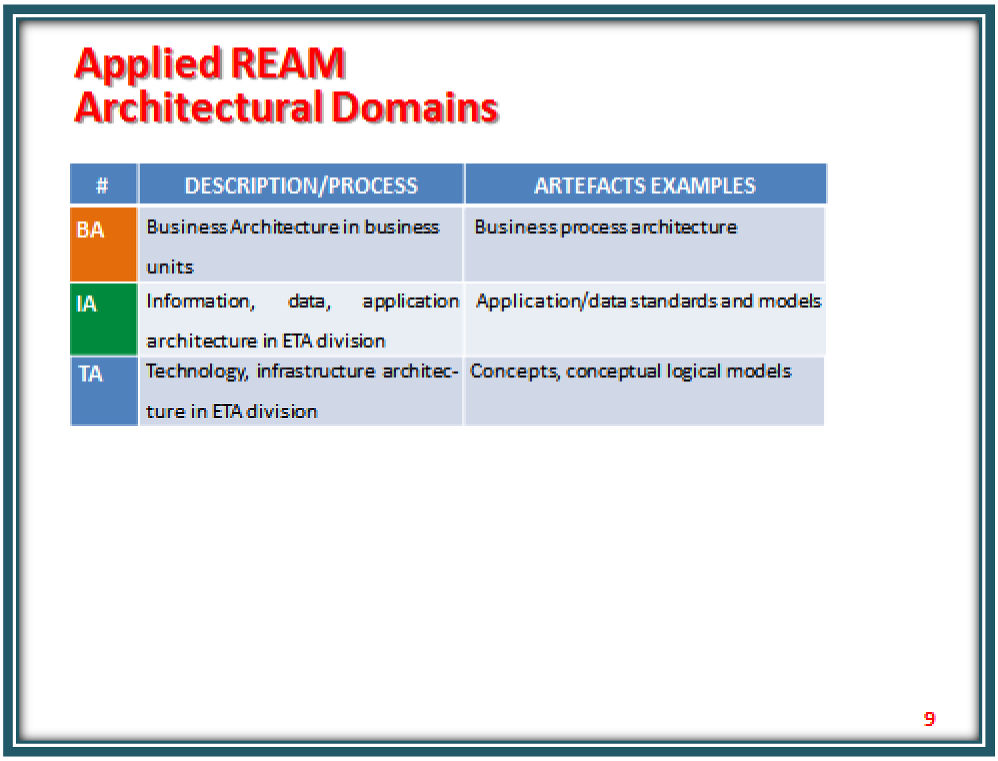


Figure A:10.1-2: Slide 9 – Case Study A Applied REAM: Domains

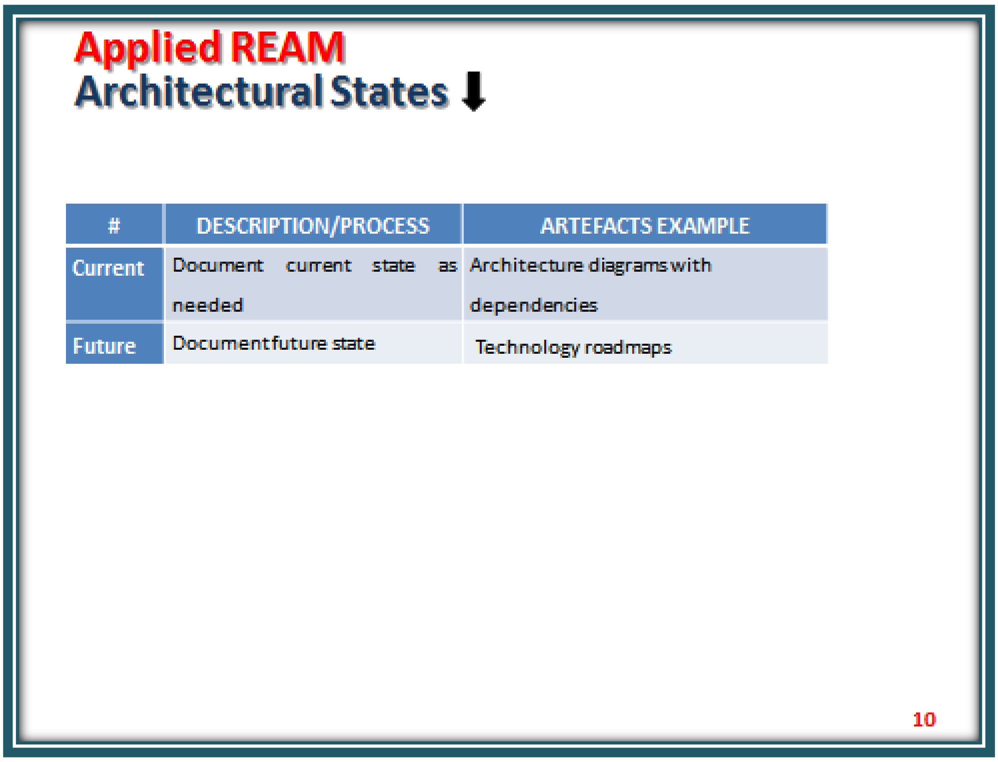


Figure A:10.1-3: Slide 10 – Case Study A Applied REAM: States

Applied REAM
Architectural Views

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
View A	Different versions according to audience as needed	Business audience brief, technical audience brief
View B	Security view	Security in operations, architecture, design, fraud and risk

11

Figure A:10.1-4: Slide 11 – Case Study A Applied REAM: Views

Applied REAM
Interlinking

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
	Matrix teams, individual knowledge	Defined roles,
	Project Management Process	Business case inputs, technology roadmaps, stage gates sign-offs, solution architectures, reference architectures
	Integrate EA into other organisational processes	Technology plan, technology delivery process, capacity projections
	Repositories of artefacts	Standalone array of tools

12

Figure A:10.1-5: Slide 12 – Case Study A Applied REAM: Interlinking

Applied REAM
Solution Architecture

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
S1	Multi-domain solution architecture (ETA division), including decommissioning	Release domain scope, security architecture, integration architecture, technical solution specification

13

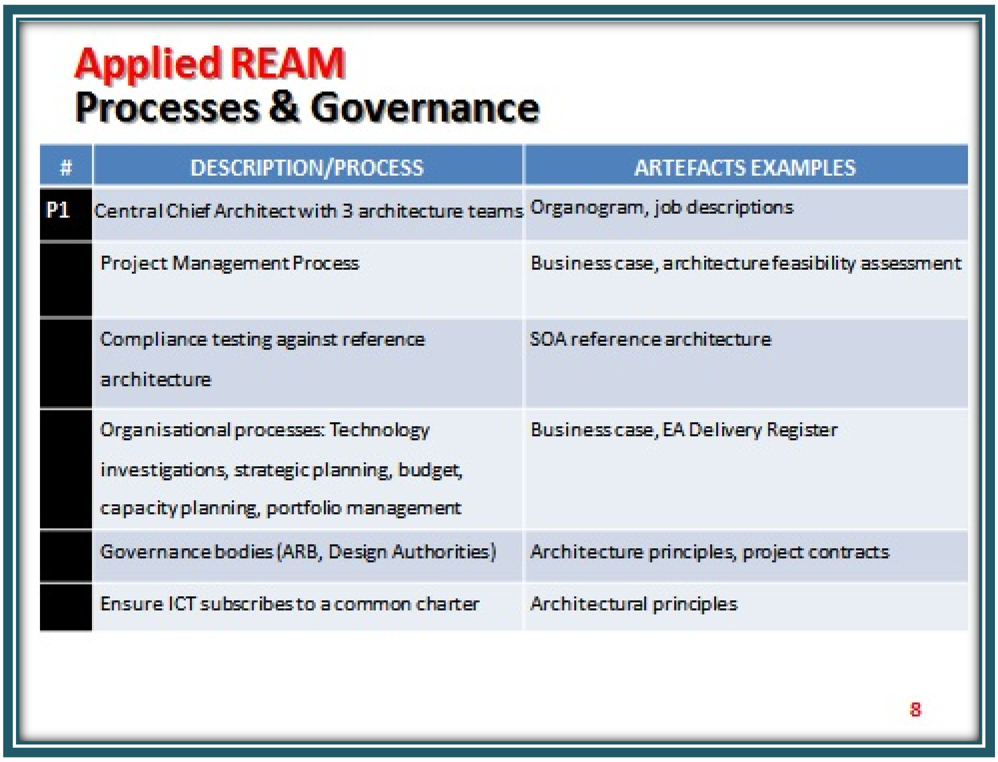
Figure A:10.1-6: Slide 13 – Case Study A Applied REAM: Solution Architecture

ADDENDUM A:11.1 SECOND SESSION – APPLICATION OF THE REAM

Addendum to Chapter 11

I have drawn up a table of the areas of the REAM and have plotted my understanding of your EA environment onto the table in an attempt to illustrate the possible practical value of the REAM.

- Slide 8 – P1 Processes & Governance (see Figure A:11.1-1).
- Slide 9 – Architectural domains, *i.e.* Business Architecture, Information Architecture and Technology Architecture (see Figure A:11.1-2).
- Slide 10 – Architectural States (see Figure A:11.1-3).
- Slide 11 – Architectural Views (see Figure A:11.1-4).
- Slide 12 – Interlinking Mechanisms (see Figure A:11.1-5).
- Slide 13 – S1 Solution Architecture (see Figure A:11.1-6).



#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLES
P1	Central Chief Architect with 3 architecture teams	Organogram, job descriptions
	Project Management Process	Business case, architecture feasibility assessment
	Compliance testing against reference architecture	SOA reference architecture
	Organisational processes: Technology investigations, strategic planning, budget, capacity planning, portfolio management	Business case, EA Delivery Register
	Governance bodies (ARB, Design Authorities)	Architecture principles, project contracts
	Ensure ICT subscribes to a common charter	Architectural principles

Figure A:11.1-1: Slide 8 – Case Study B Applied REAM: Processes & Governance

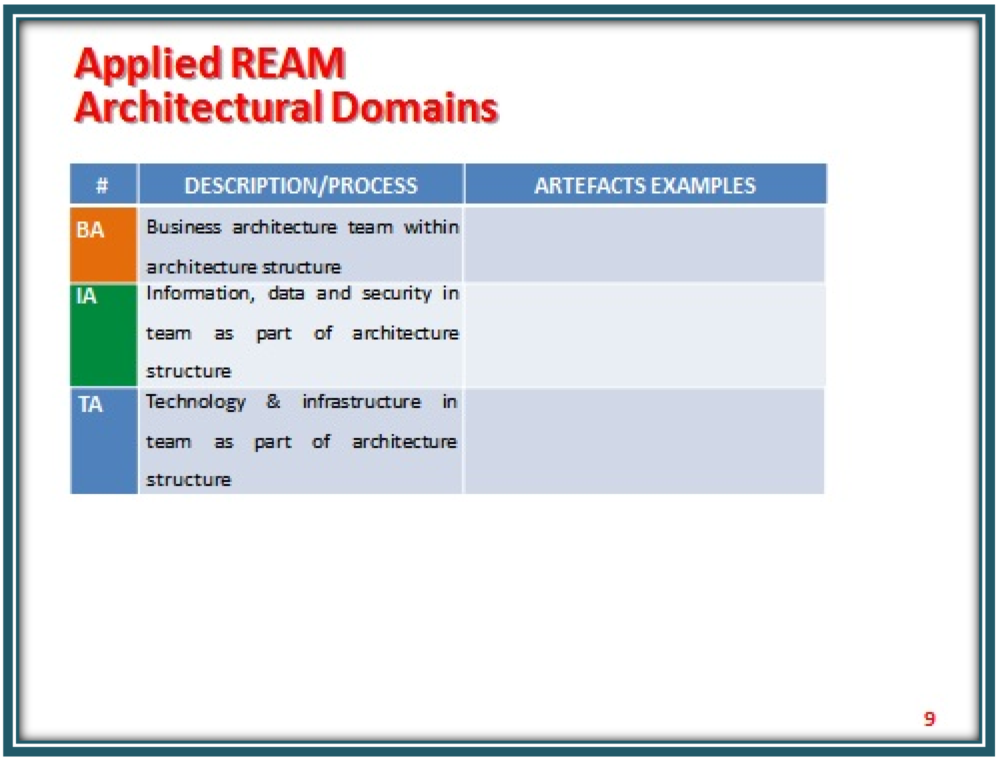


Figure A:11.1-2: Slide 9 – Case Study B Applied REAM: Domains

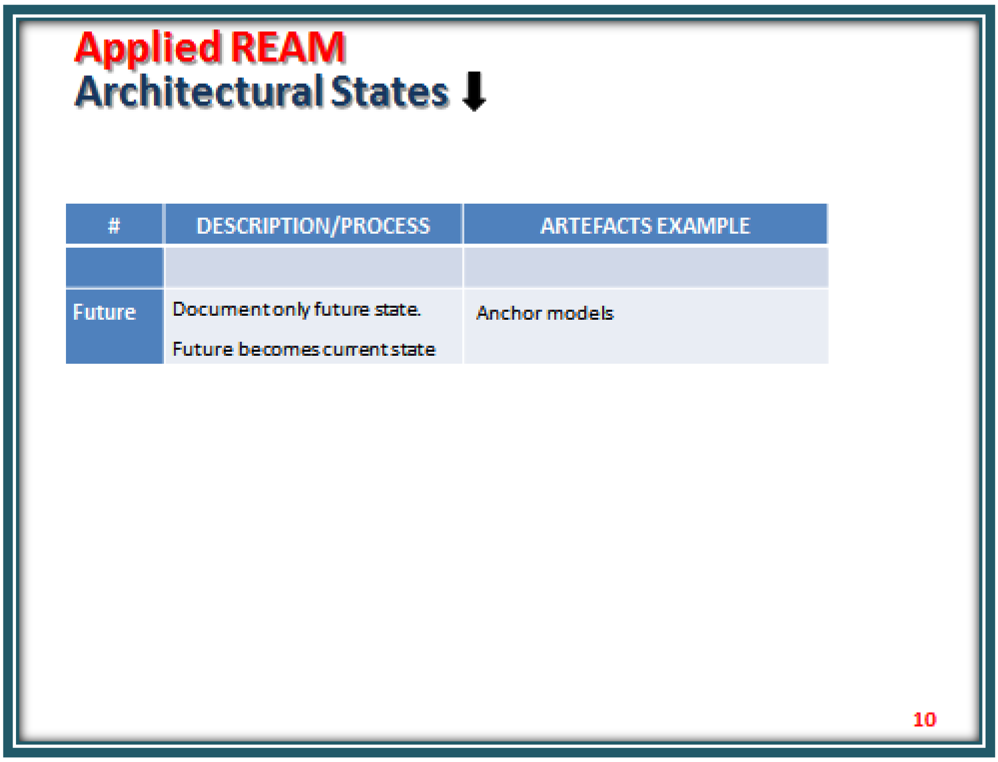
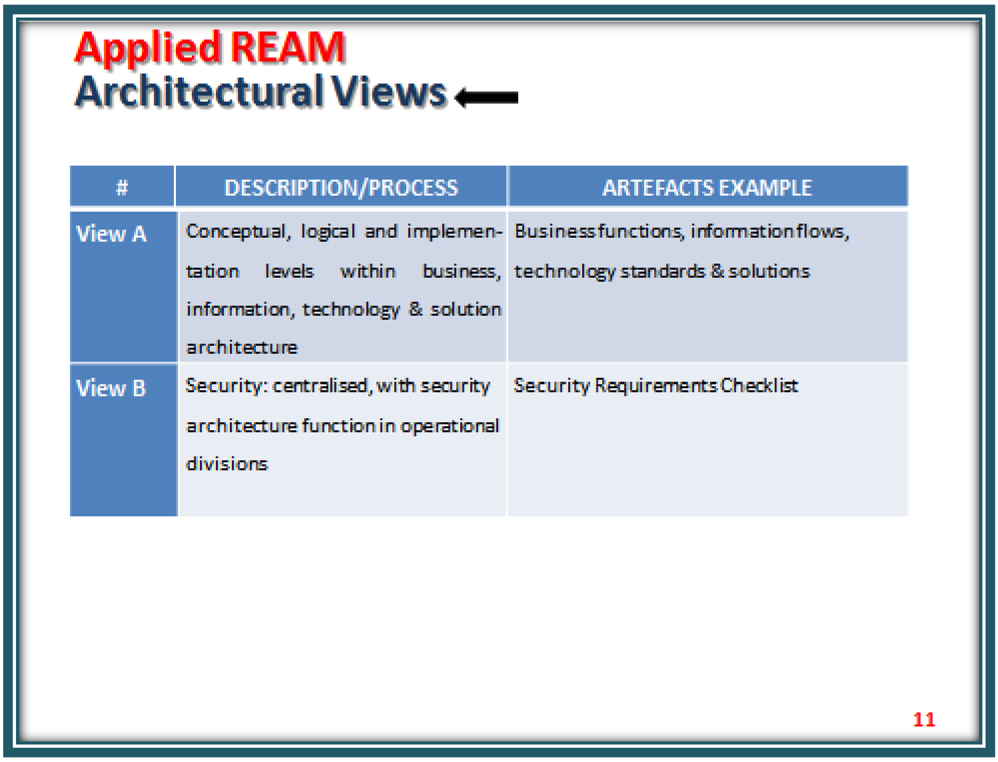


Figure A:11.1-3: Slide 10 – Case Study B Applied REAM: States

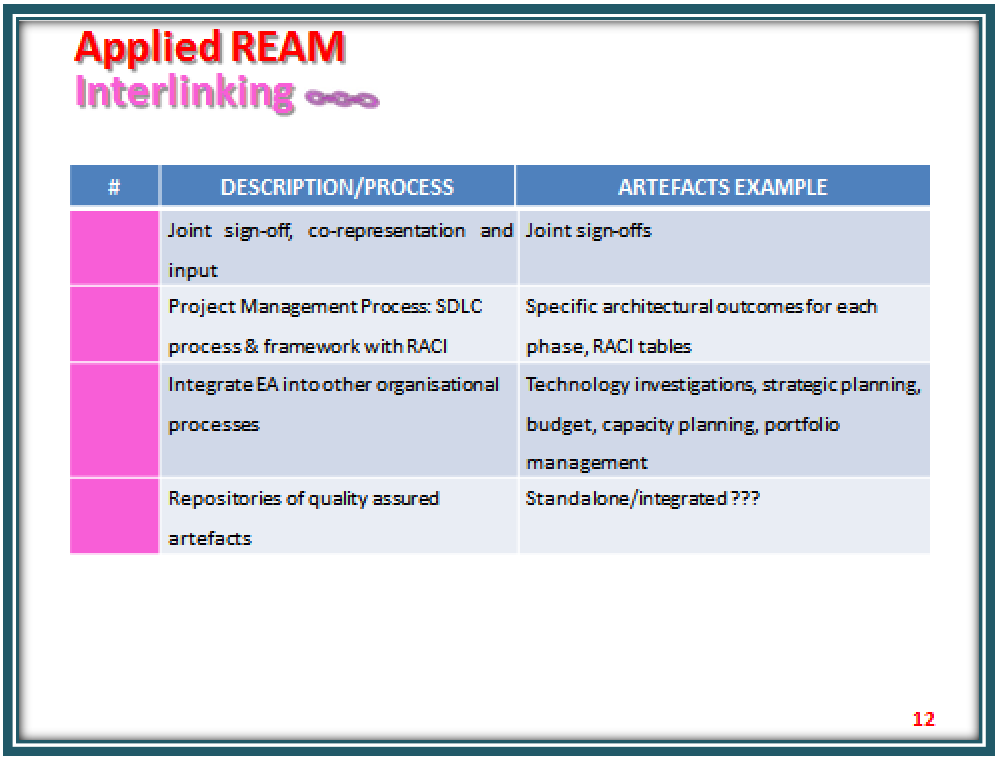


Applied REAM
Architectural Views

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
View A	Conceptual, logical and implementation levels within business, information, technology & solution architecture	Business functions, information flows, technology standards & solutions
View B	Security: centralised, with security architecture function in operational divisions	Security Requirements Checklist

11

Figure A:11.1-4: Slide 11 – Case Study B Applied REAM: Views



Applied REAM
Interlinking

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
	Joint sign-off, co-representation and input	Joint sign-offs
	Project Management Process: SDLC process & framework with RACI	Specific architectural outcomes for each phase, RACI tables
	Integrate EA into other organisational processes	Technology investigations, strategic planning, budget, capacity planning, portfolio management
	Repositories of quality assured artefacts	Standalone/integrated ???

12

Figure A:11.1-5: Slide 12 – Case Study B Applied REAM: Interlinking

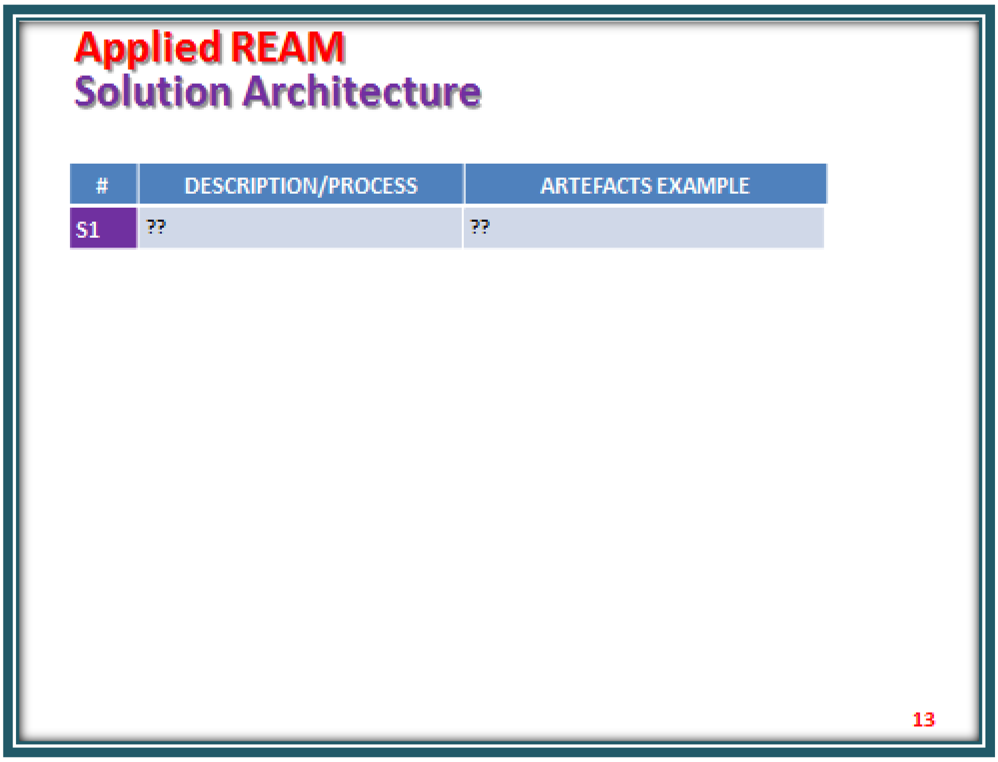


Figure A:11.1-6: Slide 13 – Case Study B Applied REAM: Solution Architecture

ADDENDUM A:12.1 SECOND SESSION – APPLICATION OF THE REAM

Addendum to Chapter 12

I have drawn up a table of the areas of the REAM and have plotted my understanding of your EA environment onto the table in an attempt to illustrate the possible practical value of the REAM.

- Slide 8 – P1 Processes & Governance (see Figure A:12.1-1).
- Slide 9 – Architectural domains, *i.e.* Business Architecture, Information Architecture and Technology Architecture (see Figure A:12.1-2).
- Slide 10 – Architectural States (see Figure A:12.1-3).
- Slide 11 – Architectural Views (see Figure A:12.1-4).
- Slide 12 – Interlinking Mechanisms (see Figure A:12.1-5).
- Slide 13 – Solution Architecture (see Figure A:12.1-6).

Applied REAM
Processes & Governance

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLES
P1	Managed in IT Solutions via IT Strategy	
	Solution Value Chain in project lifecycle	Deliverables according to phase
	Reference architecture is embedded in standards, templates, SVC	
	Organisational processes: strategic planning, budgeting, procurement, capacity planning	Minutes of approvals
	Demand management, change control board, global design authority	
	Measure alignment with strategic direction	Architecture principles

8

Figure A:12.1-1: Slide 8 – Case Study C Applied REAM Processes & Governance

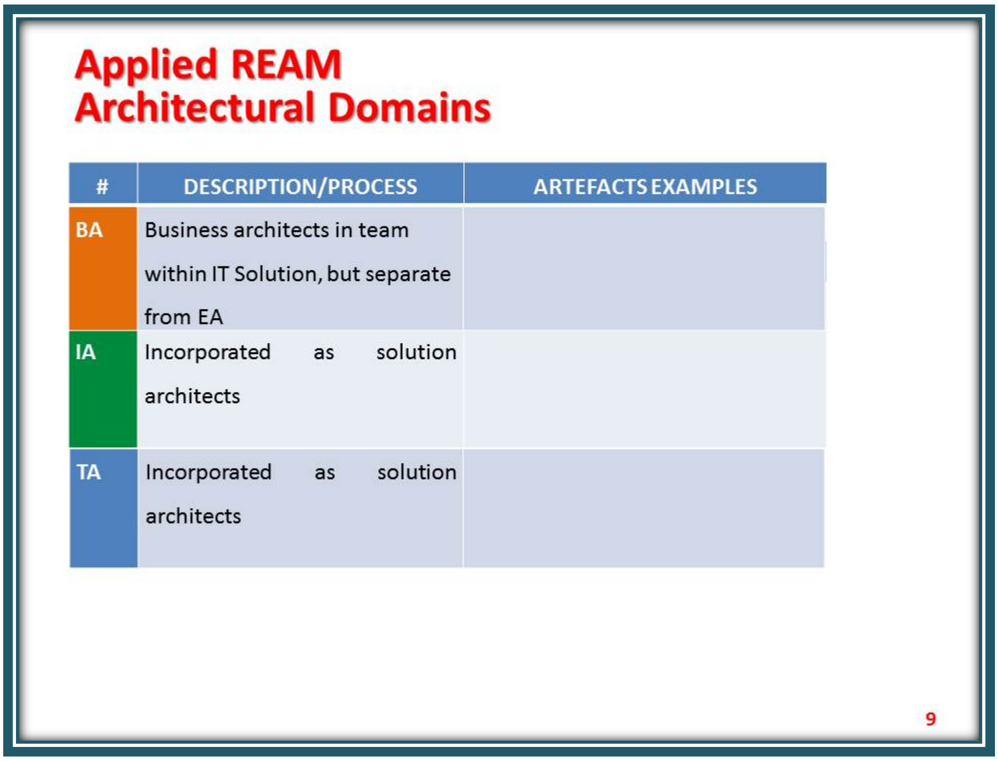


Figure A:12.1-2: Slide 9 – Case Study C Applied REAM: Domains

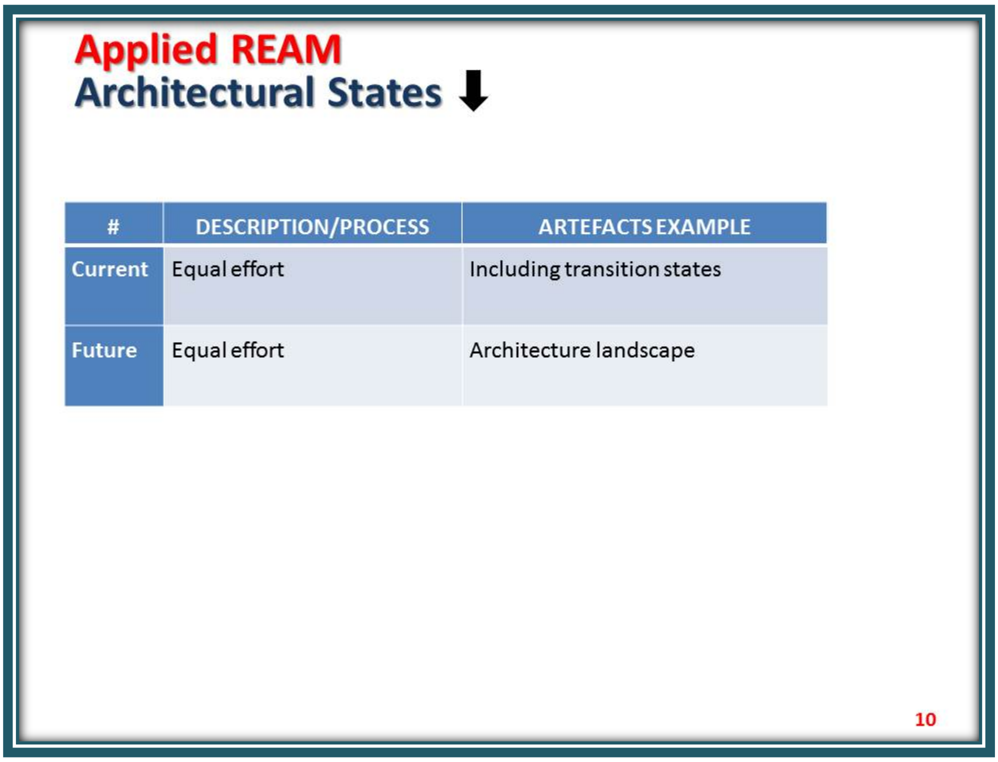


Figure A:12.1-3: Slide 10 – Case Study C Applied REAM: States

Applied REAM
Architectural Views

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
View A	Archimate views: architecture stakeholder, business stakeholder, investment stakeholder, business to business stakeholder, etc.	Documented up front
View B	Security	Security specialists in Solution Design team

11

Figure A:12.1-4: Slide 11 – Case Study C Applied REAM: Views

Applied REAM
Interlinking

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
	Joint scoping & sign-off of projects between SA & BA. Inputs into business cases. Solution architects.	
	Project Management: SVC in lifecycle	
	Input into other organisational processes - major role in strategy	
	Standalone array of tools to store artefacts	

12

Figure A:12.1-5: Slide 12 – Case Study C Applied REAM: Interlinking

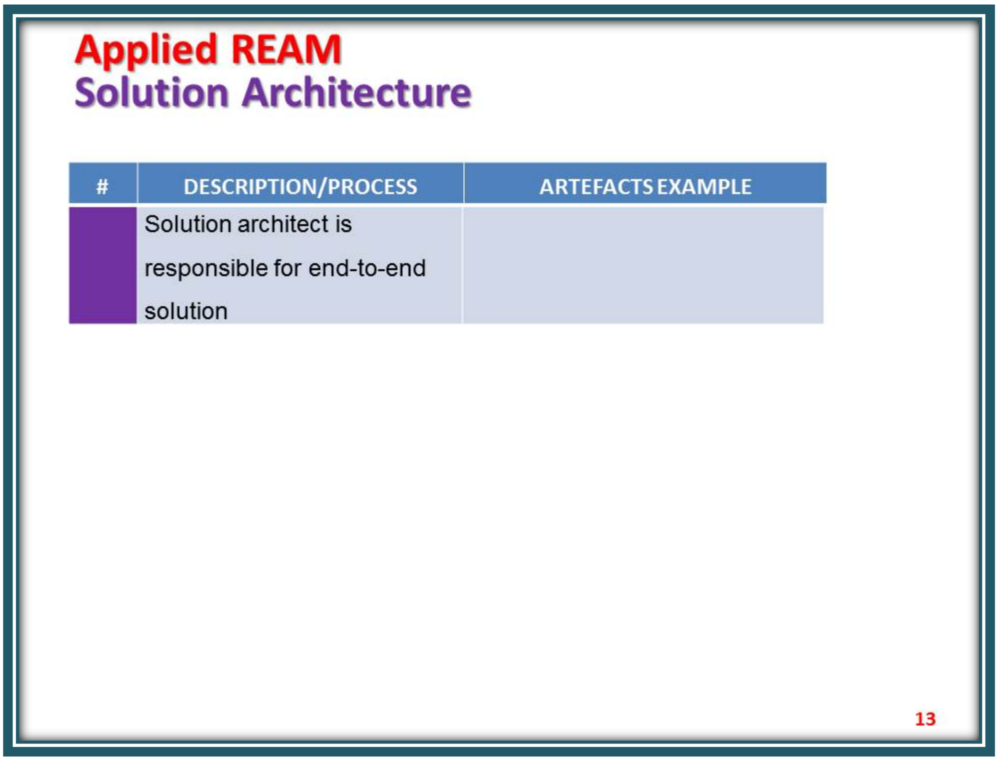


Figure A:12.1-6: Slide 13 – Case Study C Applied REAM: Solution Architecture

ADDENDUM A:14.1 BACKGROUND SUMMARY (UP)

Addendum to Chapter 14

Broadly, the research consists of three parts:

- The first part of the research (critical non-empirical study of literature) contains the history of EA and the definitions, role, functions, existing frameworks, models and ontologies of each of enterprise architecture, business architecture (BA), information architecture (IA) and technology architecture (TA). The depictions of and relationships between the different layers/domains of EA were also investigated. The outcome is, among others, a set of newly compiled definitions, of which I have provided here the definition of EA in Figure A:14.1-1, the relating of EA domains in Figure A:9.2-2 and an ontology in Figure A:9.2-3. This terminology will be adhered to in the case study process.

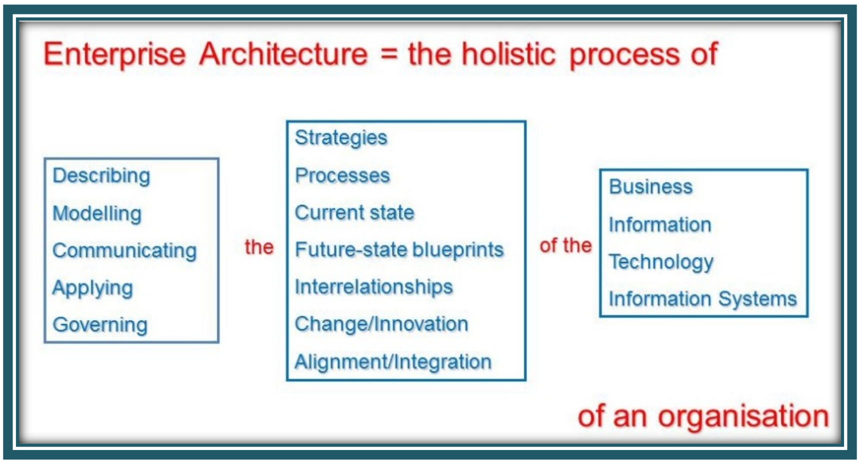


Figure A:14.1-1: Definition of Enterprise Architecture

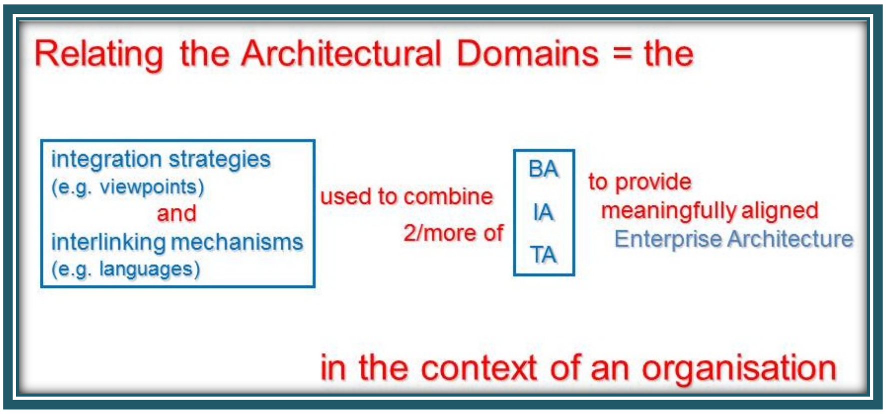


Figure A:14.1-2: Definition Relating the Architectural Domains

Figure A:14.1-3: Sample Ontology

- The second part of the research (model building) contains the design methodology and process, the design of a metamodel, the description and depiction of the metamodel, namely the Relational Enterprise Architecture Metamodel (REAM). The focus of the REAM is on strengthening the interlinking and integration between the different architectural domains.
- The third part of the research (empirical case studies) comprises the methodology and design of the research, the execution of the case studies, the cross-case analyses and the enhanced proposed model.

The purpose of the case studies was to determine the **practical application possibilities** of the proposed integration model within organisations, the possible **shortcomings** of the proposed model, and the **modifications** necessary.

Case Studies were done in three organisations with large, relatively mature EA groups in three different vertical industries. The outputs of these case studies were used to enhance and modify the proposed integration model.

The final part of the research is to determine the applicability of the proposed integration model at the University of Pretoria, based on the same methodology followed during the case studies:

- A first contact session to gather information regarding the EA function at the University of Pretoria;
- Analysis of the gathered data and a conceptual application of the proposed model at the University of Pretoria
- A second session to explain the proposed model and conceptual application thereof and to receive feedback.

ADDENDUM A:14.2 SECOND SESSION INTRODUCTION (UP)

Addendum to Chapter 14

I have had a session with you to gain an overview of your architecture environment. I have attempted to apply it conceptually to my proposed model, called the REAM – see Figure A:14.2-1. The purpose of this session is to obtain your input and expert opinion on the **practical application possibilities** of the proposed model, the possible **shortcomings** of the proposed model, and the **modifications** necessary to improve it. The intent is thus not to change, evaluate or criticise your way of doing EA (see Figure A:9.5-2 for background slide).

Figure A:14.2-1: UP: Introduction Slide

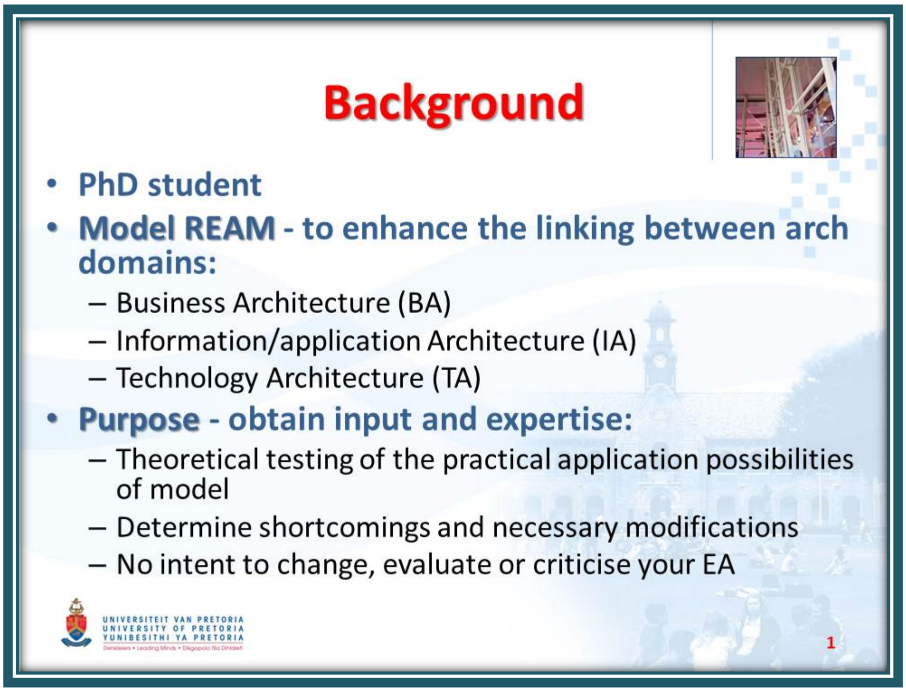


Figure A:14.2-2: UP Slide 1 – Background

First, I will explain the basic model

- Slide 2 – The REAM is based on a three-dimensional hexagon drawn as a geometrical net to allow better visibility (see Figure A:9.5-3).
- Slide 3 – On the first dimension, the architectural domains are displayed (see Figure A:9.5-4).
- Slide 4 – The second dimension addresses the current and future-state architectures (see Figure A:9.5-5).
- Slide 5 – The third dimension enables the capability to address different architectural views (see Figure A:9.5-6).
- Slide 6 – The last dimension adds the relations (Figure A:9.5-7).
- Slide 7 – Each area is numbered for easier reference (Figure A:9.5-8).
- Slide 8 – The final combined metamodel (Figure A:14.2-9).

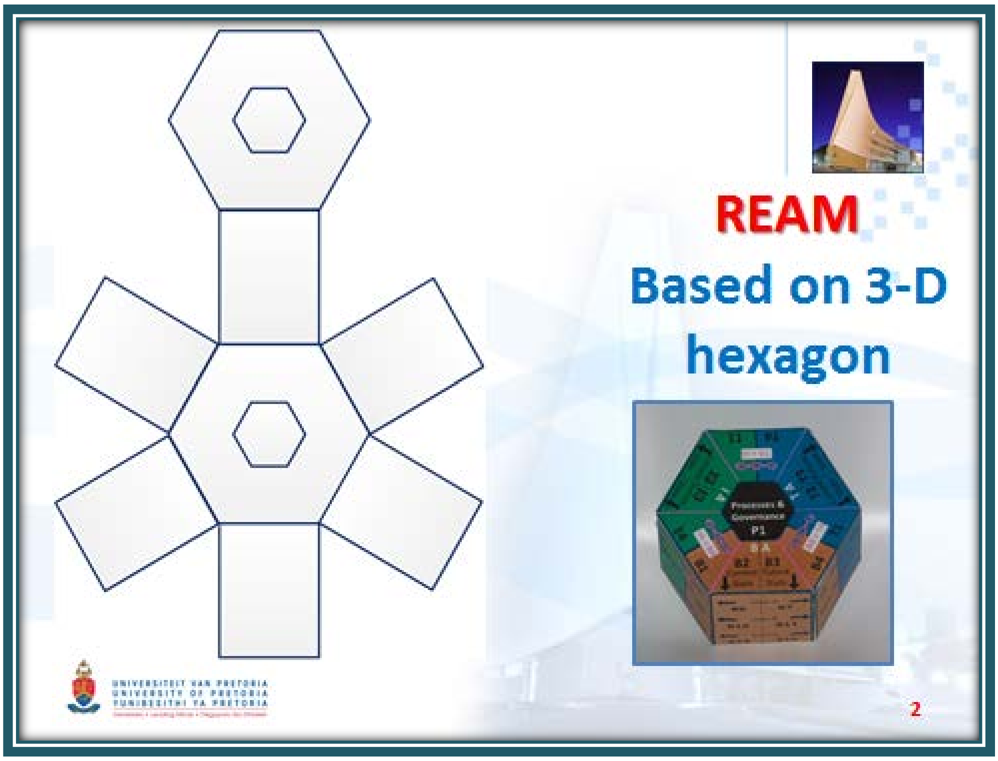


Figure A:14.2-3: Slide 2 – The Modified REAM

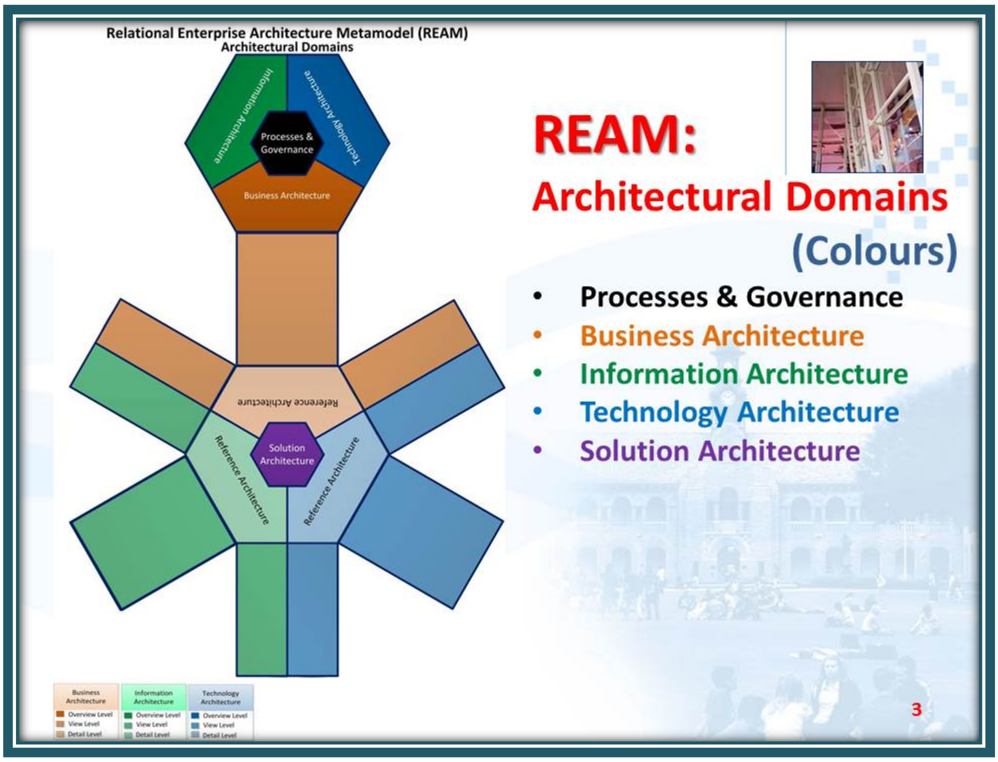


Figure A:14.2-4: Slide 3 – The Modified REAM

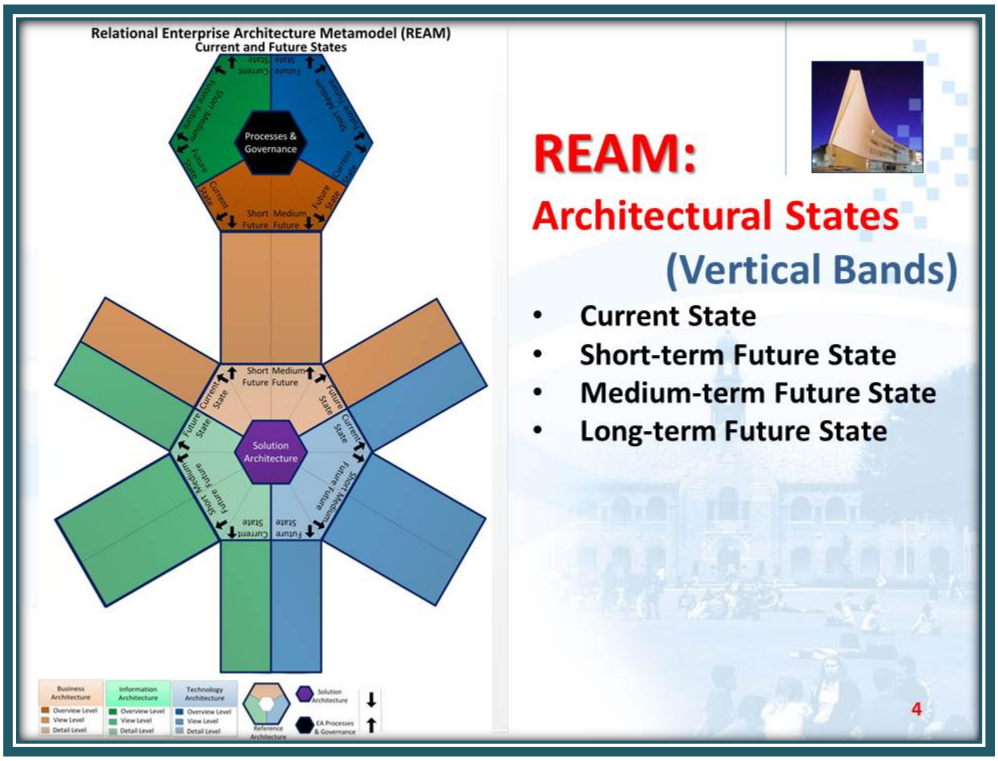


Figure A:14.2-5: Slide 4 – The Modified REAM

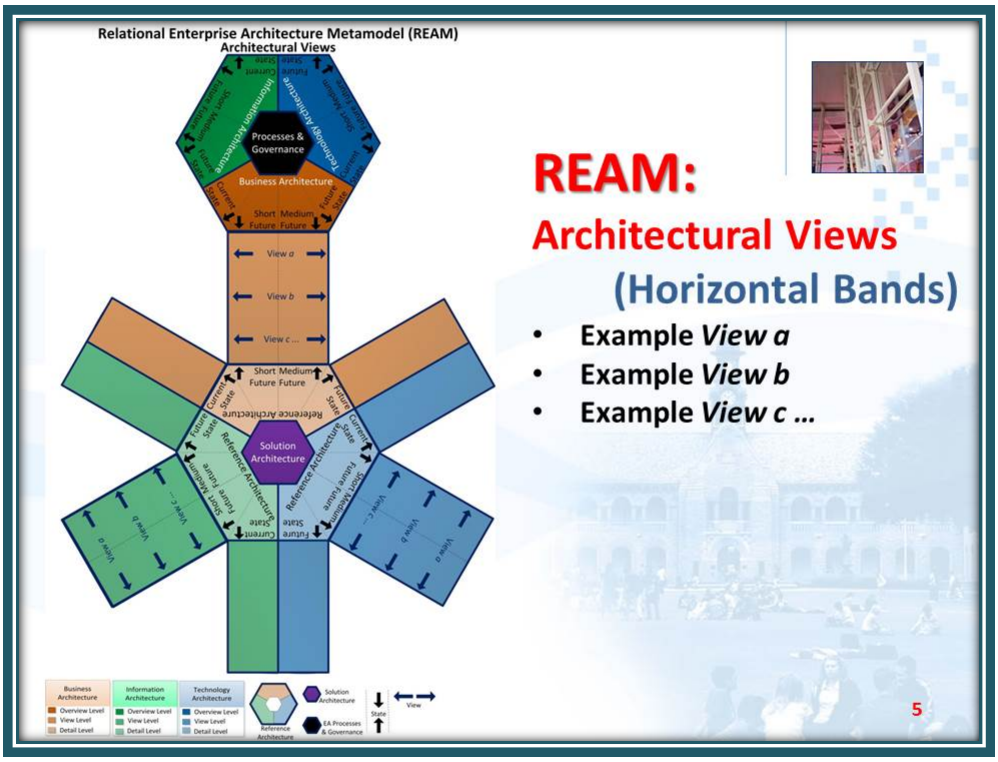


Figure A:14.2-6: Slide 5 – The Modified REAM

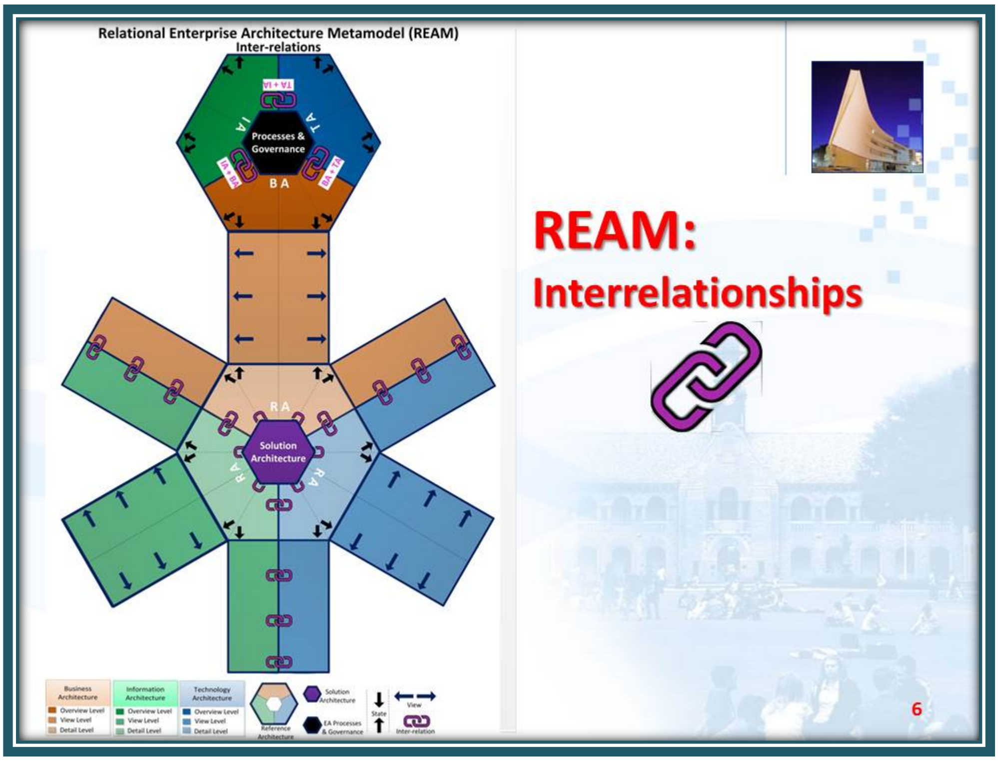


Figure A:14.2-7: Slide 6 – The Modified REAM

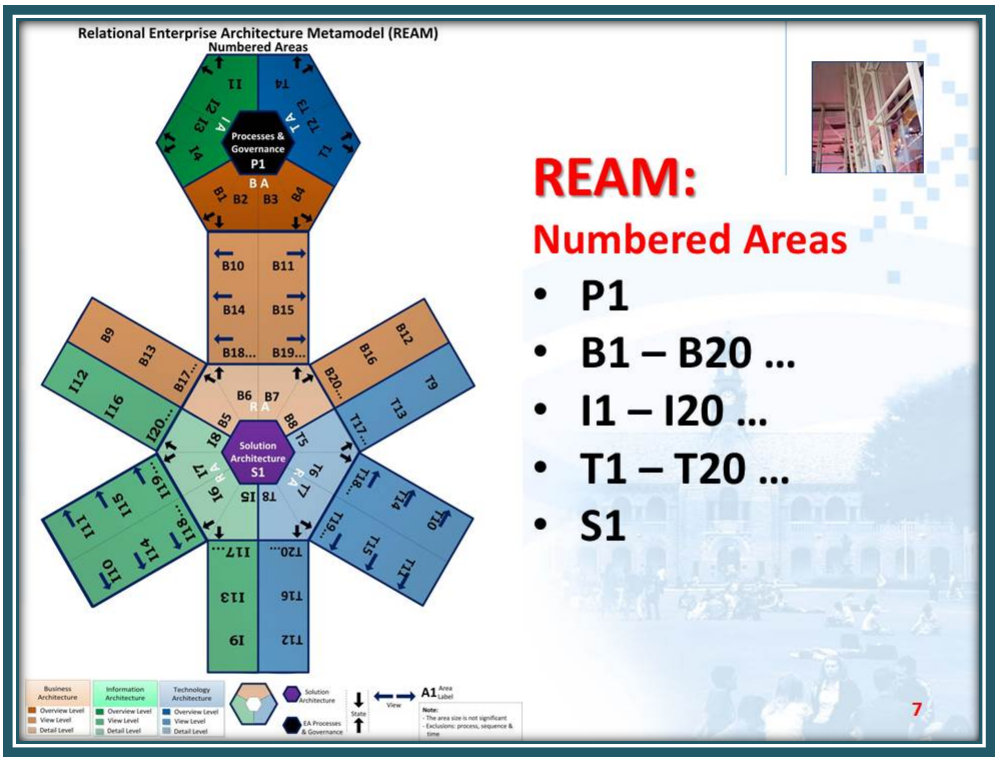


Figure A:14.2-8: Slide 7 – The Modified REAM

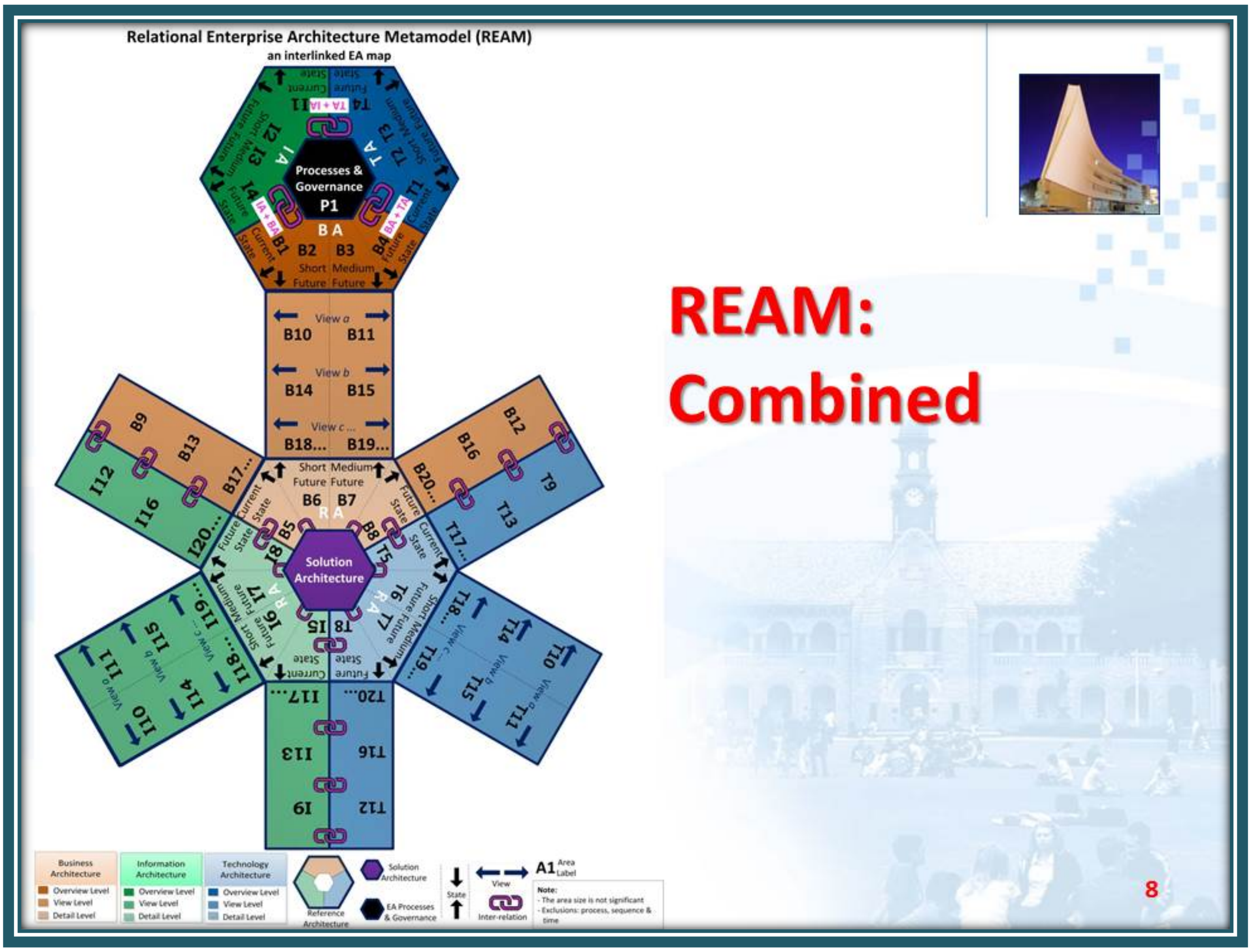


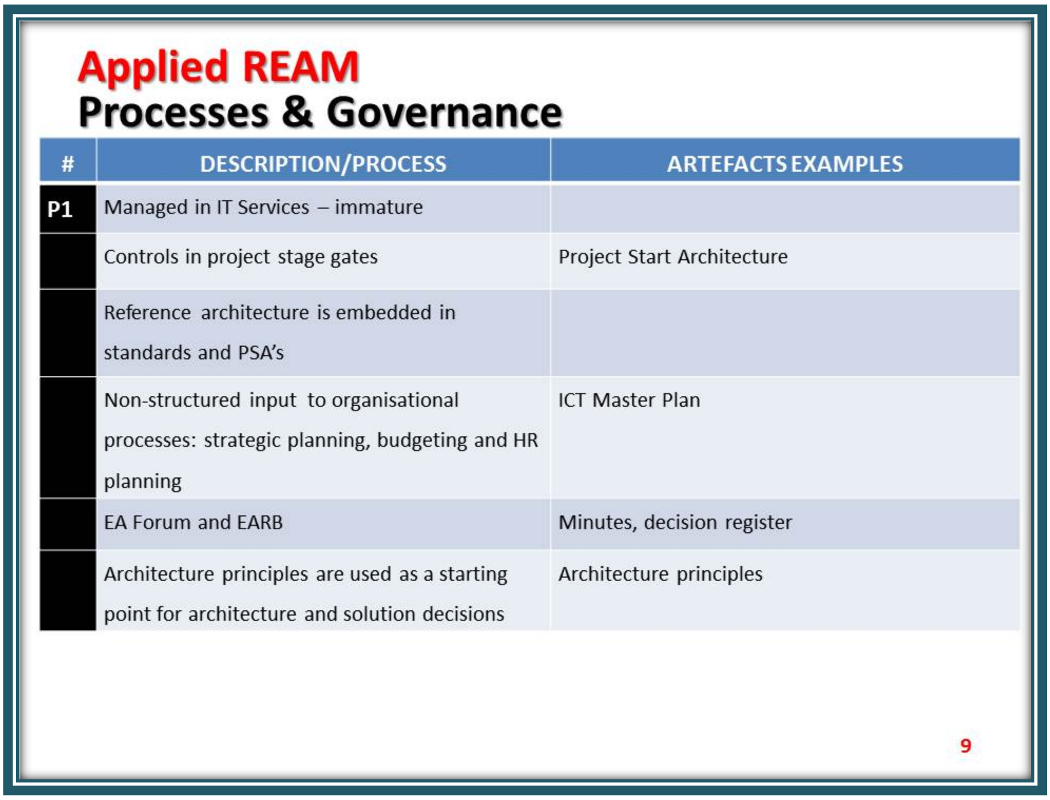
Figure A:14.2-9: Slide 8 – The Modified REAM

ADDENDUM A:14.3 SECOND SESSION – APPLICATION OF THE REAM (UP)

Addendum to Chapter 14

I have drawn up a table of the areas of the REAM and have plotted my understanding of your EA environment onto the table in an attempt to illustrate the possible practical value of the REAM.

- Slide 9 – P1 Processes & Governance (see Figure A:14.3-1).
- Slide 10 – Architectural Domains, *i.e.* business architecture, information architecture and technology architecture (see Figure A:9.5-10).
- Slide 11 – Architectural States (see Figure A:14.3-3).
- Slide 12 – Architectural Views (see Figure A:9.5-12).
- Slide 13 – Interlinking Mechanisms (see Figure A:9.5-13).
- Slide 14 – Solution Architecture (see Figure A:9.5-14).
- Slides 15 & 16 – Detailed example of area T7 and a technology roadmap (see Figure A:14.3-7 and Figure A:14.3-8).

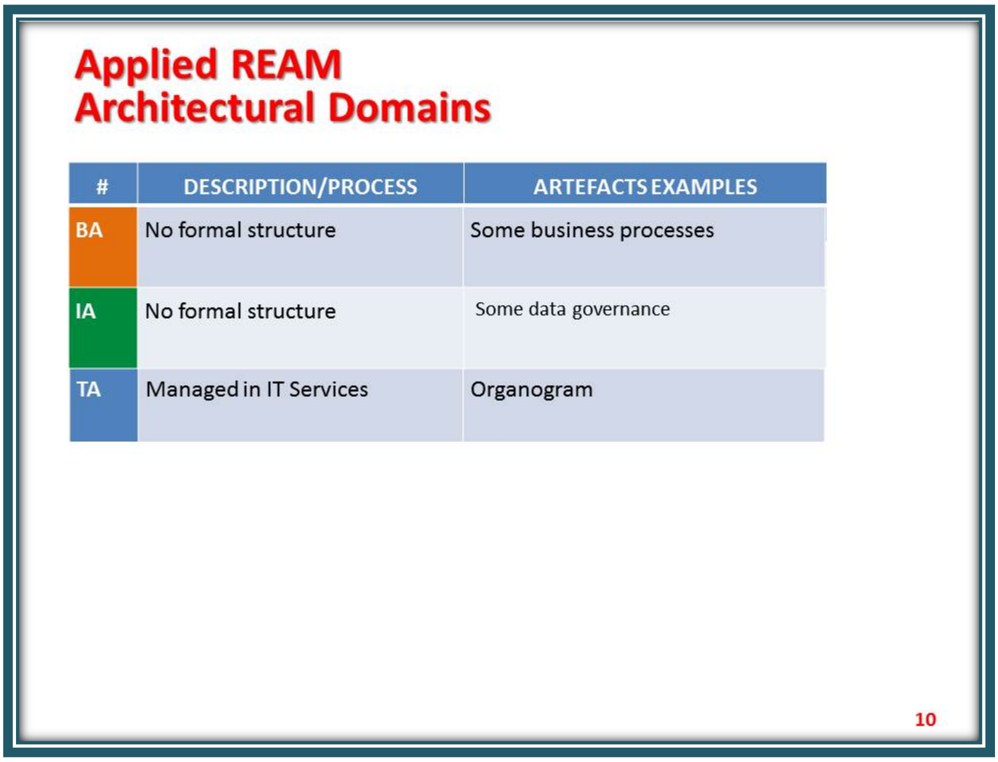


Applied REAM
Processes & Governance

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLES
P1	Managed in IT Services – immature	
	Controls in project stage gates	Project Start Architecture
	Reference architecture is embedded in standards and PSA's	
	Non-structured input to organisational processes: strategic planning, budgeting and HR planning	ICT Master Plan
	EA Forum and EARB	Minutes, decision register
	Architecture principles are used as a starting point for architecture and solution decisions	Architecture principles

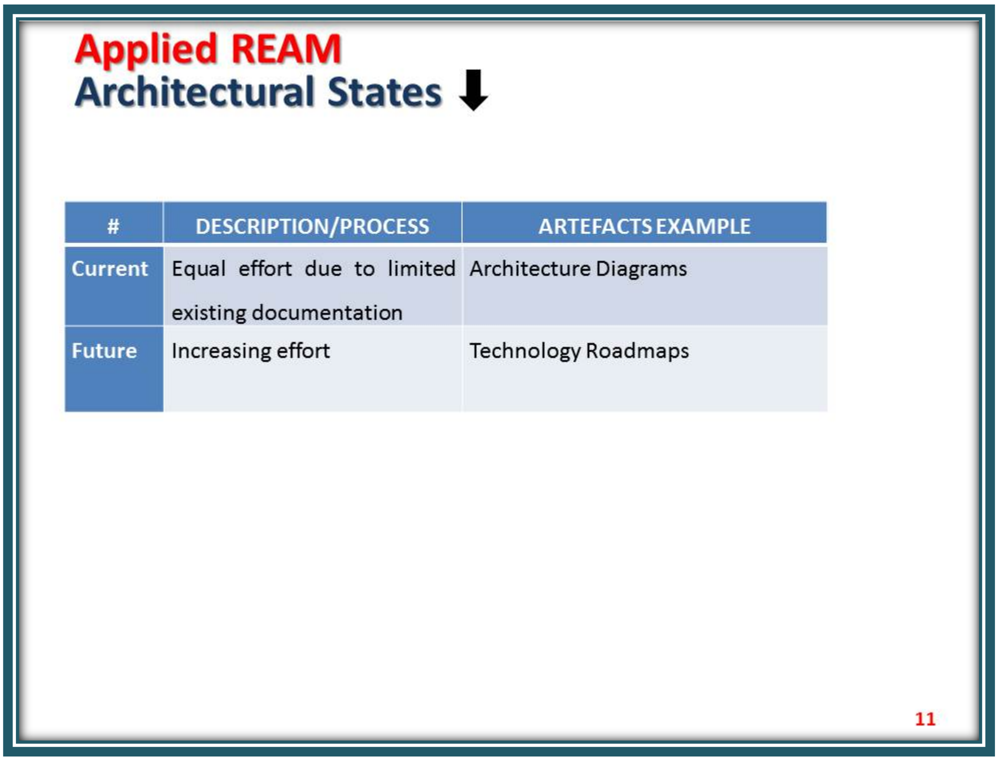
9

Figure A:14.3-1: Slide 9 – UP Applied REAM: Processes & Governance



#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLES
BA	No formal structure	Some business processes
IA	No formal structure	Some data governance
TA	Managed in IT Services	Organogram

Figure A:14.3-2: Slide 10 – UP Applied REAM: Domains



#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
Current	Equal effort due to limited existing documentation	Architecture Diagrams
Future	Increasing effort	Technology Roadmaps

Figure A:14.3-3: Slide 11 – UP Applied REAM: States

Applied REAM
Architectural Views

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
View a	Different viewpoints will be developed as required	Executive proposal
View b	Security is handled within each team.	
View c …		

12

Figure A:14.3-4: Slide 12 – UP Applied REAM: Views

Applied REAM
Interlinking

#	DESCRIPTION/PROCESS	ARTEFACTS EXAMPLE
	Interlinking is included within artefacts	Roadmaps
	Project Methodology	PSA
	Input into other organisational processes	
	Oracle WebCenter Spaces store artefacts	

13

Figure A:14.3-5: Slide 13 – UP Applied REAM: Interlinking

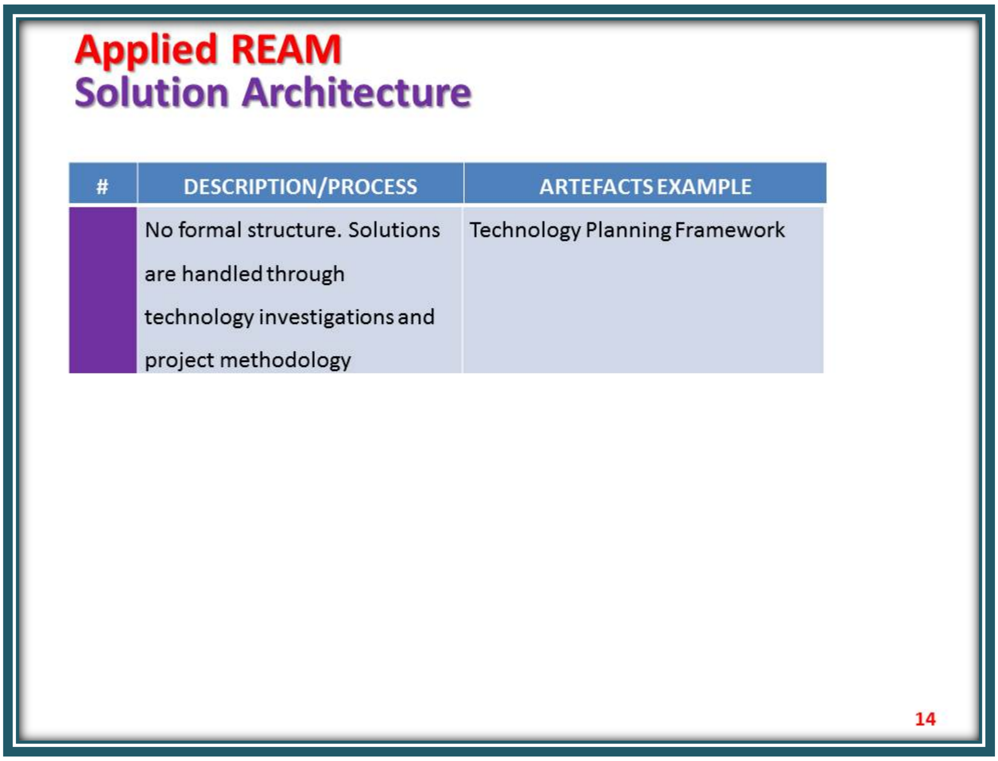


Figure A:14.3-6: Slide 14 – UP Applied REAM: Solution Architecture

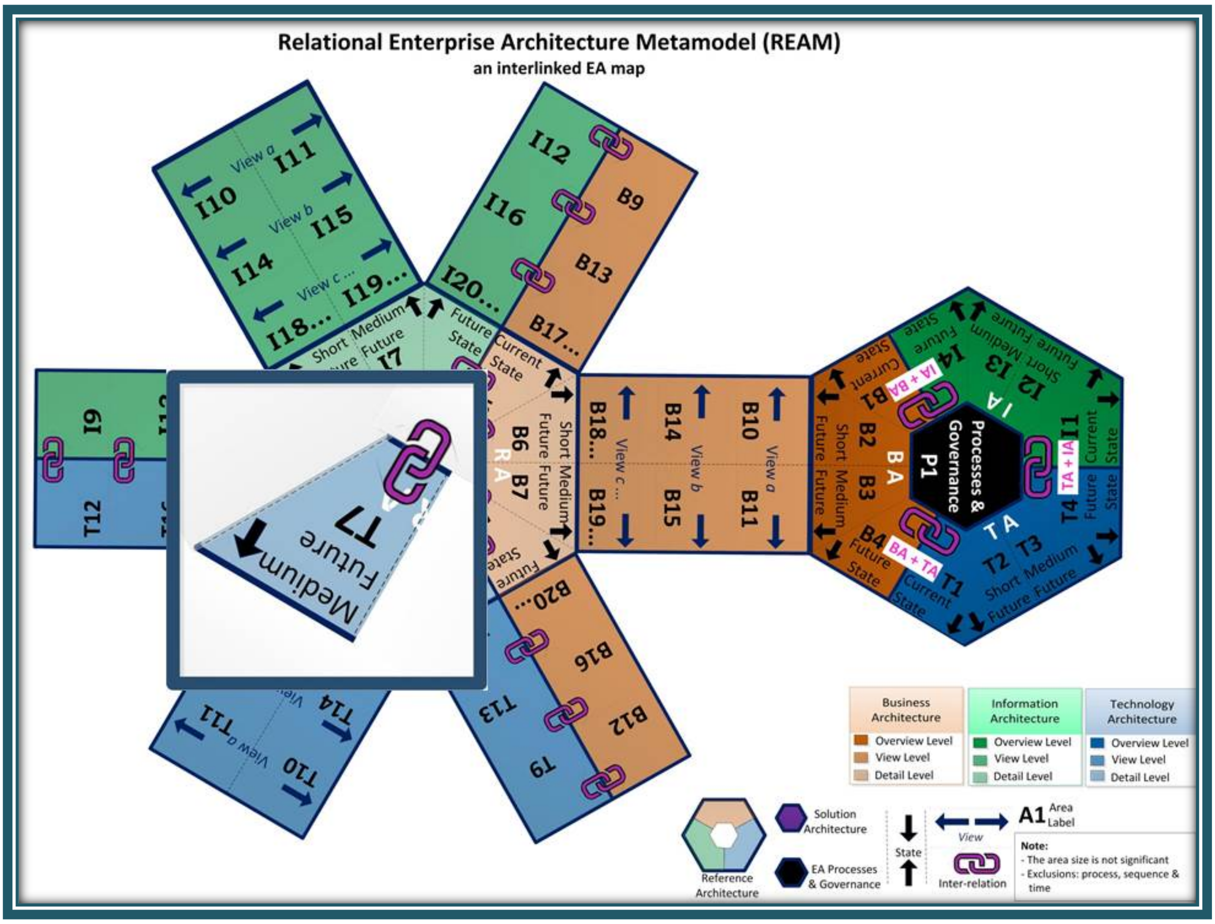
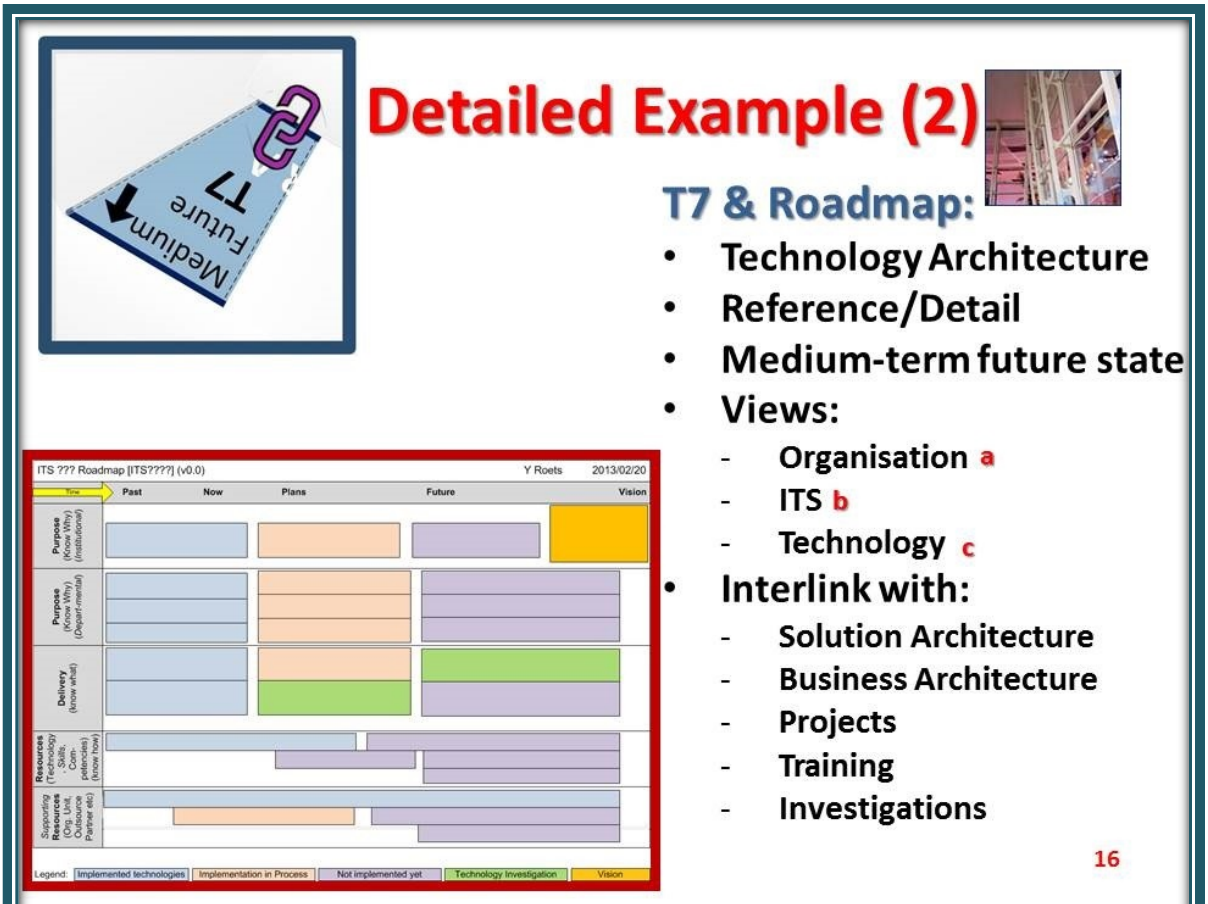


Figure A:14.3-7: Slide 15 – UP Applied REAM: Detailed Example



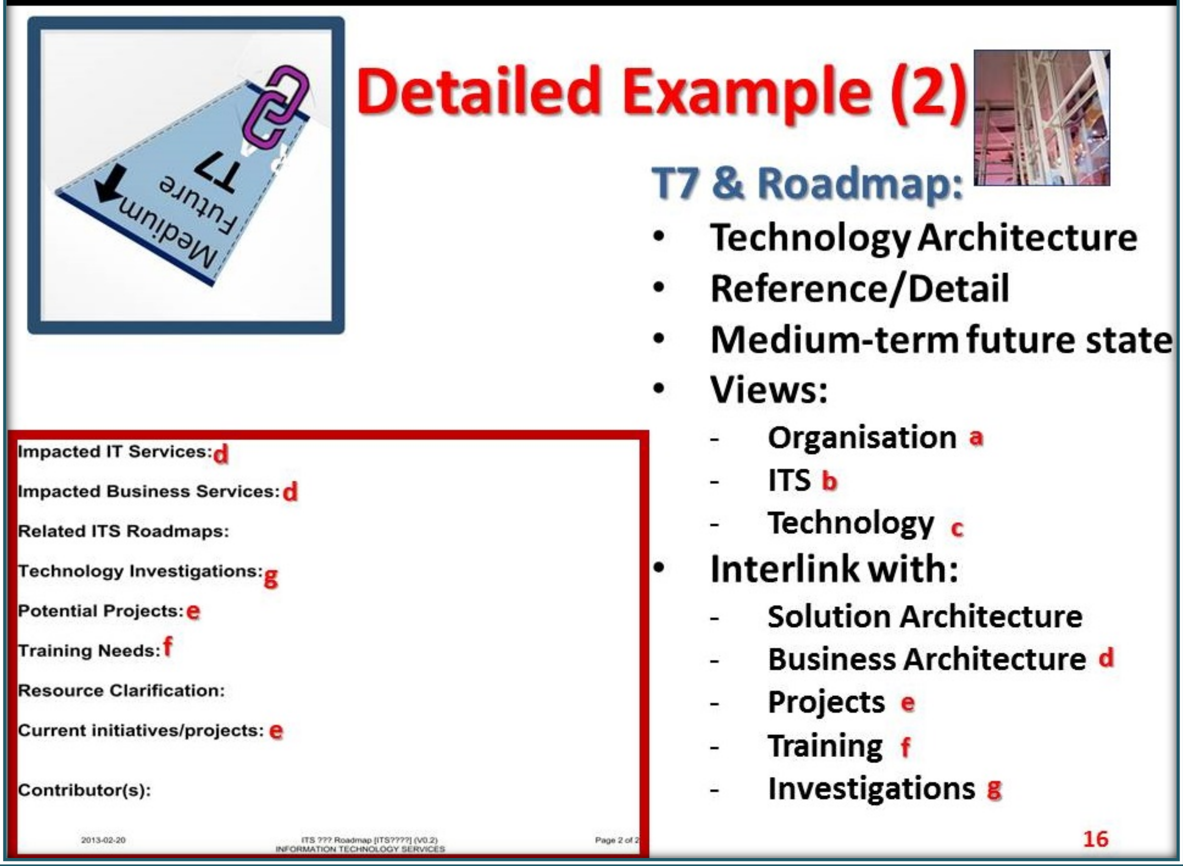


Figure A:14.3-8: Slide 16 – UP Applied REAM: Area T7 and Roadmaps

REFERENCES

Abdallah, S. & Galal-Edeen, G.H. 2006. *Towards a Framework for Enterprise Architecture Frameworks Comparison and Selection*. Paper presented at the Fourth International Conference on Informatics and Systems, Cairo. 10p. [Online]. Available: <http://ggatz.com/images/ISA_frameworks.pdf> [Accessed 5 February 2011].

Addison, W. 2007. *St.George Bank: IT Architecture Model*. St George Bank. 6p. [Online]. Available: <http://www.sparxsystems.com.au/downloads/pdf/st_george_bank_case_study.pdf> [Accessed 26 March 2012].

Aier, S., Kurpjuweit, S., Saat, J. & Winter, R. 2009. Enterprise Architecture Design as an Engineering Discipline. *AIS Transactions on Enterprise Systems,* 1(1): 36-43. [Online]. Available: <www.alexandria.unisg.ch/export/DL/204778.pdf> [Accessed 14 November 2011].

Aier, S. & Schönherr, M. 2006. Evaluating Integration Architectures - A Scenario-based Evaluation of Integration Technologies. *Lecture Notes in Computer Science,* 3888: 2-14. [Online]. Available: <http://www.martenschoenherr.de/pdf/VLDB_EvaluatingIntegrationArchitectures_abstract.pdf> [Accessed 20 February 2011].

Aier, S. & Winter, R. 2009. Virtual Decoupling for IT/Business Alignment - Conceptual Foundations, Architecture Design and Implementation Example. *Business & Information Systems Engineering,* 1(2): 150-163.

Akhavan, P., Jafari, M. & Ali-Ahmad, A.R. 2006. Exploring the Interdependency between Reengineering and Information Technology by Developing a Conceptual Model. *Business Process Management Journal,* 12(4): 517-534.

Alonso, I.A., Verdún, J.C. & Caro, E.T. 2010. *The IT Implicated within the Enterprise Architecture Model: Analysis of Architecture Models and Focus IT Architecture Domain*. Paper presented at the IEEE International Conference on Service-Oriented Computing and Applications (SOCA). 1-5p. [Online]. Available:

<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5707174> [Accessed 15 May 2011].

Alter, S. 2005. Architecture of Sysperanto: a Model-based Ontology of the IS Field. *Communications of the Association for Information Systems,* 15: 1-40. [Online]. Available: <http://web.ebscohost.com/ehost/pdfviewer/pdfviewer?vid=3&hid=11&sid=79b42336-ae83-4f69-8190-b14e5237ec8e%40sessionmgr14> [Accessed 4 April 2012].

Ambler, S.W. 2009. *Architecture and Architecture Modeling Techniques*. [Online]. Available: <http://www.agiledata.org/essays/enterpriseArchitectureTechniques.html> [Accessed 27 March 2012].

Ambler, S.W. 2011. *Effective Practices for Modeling and Documentation*. [Online]. Available: <http://www.agilemodeling.com/> [Accessed 21 August 2011].

Anon. 2008a. *Business Modelling - IT and Business Alignment*. Methodologies Corporation.p. [Online]. Available: <http://www.modelingconcepts.com/pdf/BPM_V2.pdf> [Accessed 19 July 2011].

Anon. 2008b. *Enterprise Framework Standards*. [Online]. Available: <http://zachmanframeworkassociates.com/Standards/protected/framework-template?id=3> [Accessed 10 September 2011].

Anon. 2009. *Agile Business Process Modelling Framework and Enterprise Architecture*. 13p. [Online]. Available: <http://cf.agilealliance.org/articles/system/article/file/740/file.pdf> [Accessed 21 August 2011].

Anon. 2012a. *IT Architecture Modeling with MEGA Architecture*. [Online]. Available: <http://www.mega.com/en/c/product/p/modeling/p2/it-architecture-modeling> [Accessed 27 March 2012].

Anon. 2012b. *OMG's MetaObject Facility*. [Online]. Available: <http://www.omg.org/mof/> [Accessed 26 May 2012].

Armenio, F., Barthel, H., Dietrich, P., Duker, J., Floerkemeier, C., Garrett, J. & Harrison, M. 2009. *The EPCGLobal Architecture Framework*. EPCGlobal. 74p. [Online]. Available: <http://www.gs1.org/gsmp/kc/epcglobal/architecture/architecture_1_3-framework-20090319.pdf> [Accessed 12 September 2012].

Armour, F.J., Kaisler, S.H. & Liu, S.Y. 1999. A Big Picture Look at Enterprise Architectures. *IT Professional Magazine,* January/February: 35-42.

Badenhorst, C. 2008. *Dissertation writing: a research journey*. Pretoria: Van Schaik. 232p.

Ballengee, B. 2010a. Enterprise Architecture: Time for IT to Break Out! In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 147-155). Boca Raton: CRC Press.

Ballengee, B. 2010b. Why Enterprise Architecture Matters: Surfing the Waves. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 46-51). Boca Raton: CRC Press.

Bates, N. & Nelson, D. 2009. *Technology Architecture Framework* [Presentation]. Wellington: Victoria University of Wellington. [Online]. Available: <http://www.victoria.ac.nz/its/information-policies/documents/TechnologyArchitecture.pdf> [Accessed 6 January 2011].

Batley, S. 2007. *Information Architecture for Information Professionals*. Oxford: Chandos Publishing. 212p.

Baudoin, C.R. 2010. EA at 23: Allowed in the Bar, but still being Carded. *Enterprise Architecture,* 13(1): 1-23.

Bedwell, D. 2006. *Enterprise Architecture* [Presentation]. Australia: CAUDIT. [Online]. Available: <http://www.csu.edu.au/special/eac/documents/CAUDIT_Enterprise_Architecture_May_2006.pdf> [Accessed 4 Jun 2011].

Bernard, S.A. 2005. *An Introduction to Enterprise Architecture* (2nd ed.). Bloomington: AuthorHouse. 352p.

Beyer, M., Kuhn, K.A., Meier, C., Jablonski, S. & Lenz, R. 2004. *Towards a Flexible Process-Oriented IT Architecture for an Integrated Healthcare Network.* Paper presented at the ACM Symposium on Applied Computing, Nicosia. [Online]. Available: <http://mailserver.di.unipi.it/ricerca/proceedings/AppliedComputing04/Papers/T05P07.pdf> [Accessed 4 April 2012].

Bhatt, G.D. 2000. Exploring the relationship between information technology, infrastructure and business process re-engineering. *Business Process Management Journal,* 6(2): 139-163.

Blechar, M. 2009. *A Reference Architecture for Collaborative Enterprise Business Process Improvement Technologies*. (Gartner G00169319), 13p.

Blechar, M.J. 2007. *Establishing an Agile Information Architecture*. (Gartner G00152365), 9p.

Blevins, T.J., Spencer, J. & Waskiewicz, F. 2004. TOGAF ADM and MDA: the Power of Synergy. 24p. [Online]. Available: <https://www.opengroup.org/architecture/wp/ > [Accessed 5 February 2011].

Blowers, M. 2012. *Hybrid Enterprise Architecture Frameworks are in the Majority* [Blog] [Online]. <http://ovum.com/2012/03/22/hybrid-enterprise-architecture-frameworks-are-in-the-majority/> [Accessed 10 August 2012].

Boh, W. & Yellin, D. 2007. Using Enterprise Architecture Standards in Managing Technology. *Journal of Management Information Systems,* 23(3): 163-207.

Bologa, C., Faur, G. & Ghisoiu, N. 2010. Generic Model of an Enterprise Information Architecture for a Public Institution. *Journal of Computer Science and Control Systems,* 3(1): 19-24. [Online]. Available: <http://electroinf.uoradea.ro/reviste%20CSCS/volumes/JCSCS_Nr_1_integral.pdf> [Accessed 31 January 2012].

Borysowich, C. 2007. *IDEF4 Models - Object Oriented Model*. [Online]. Available: <http://it.toolbox.com/blogs/enterprise-solutions/idef4-models-object-oriented-model-18724> [Accessed 19 July 2011].

Bredemeyer, D. & Malan, R. 2001. *Architecture Definitions* [Presentation]: Bredemeyer Consulting. [Online]. Available: <http://www.bredemeyer.com/pdf_files/Definitions.pdf> [Accessed 20 February 2012].

Brinck, T., Gergle, D. & Wood, S.D. 2001. *Usability for the Web: Designing Web Sites that Work*. San Francisco: Morgan Kaufmann. 320p.

Brown, A.W. 2008. *MDA redux: Practical realization of model driven architecture.* Paper presented at the 7th International Conference on Composition-Based Software Systems, Madrid. [Online]. Available: <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4463986> [Accessed 16 March 2012].

Bruls, W.A.G., Van Steenbergen, M., Foorthuis, R.M., Bos, R. & Brinkkemper, S. 2010. Domain Architectures as an Instrument to Refine Enterprise Architecture. *Communications of the Association for Information Systems,* 27(1): 517-540. [Online]. Available: <http://aisel.aisnet.org/cais/vol27/iss1/27> [Accessed 20 Feb 2011].

Burgstahler, S. 2012. *Universal Design: Process, Principles, and Applications*. [Online]. Available: <http://www.washington.edu/doit/Brochures/PDF/ud.pdf> [Accessed 16 August 2012].

Burton, B. 2009. *Enterprise Architecture Seminar Workshop Results: Top EA Challenges*. (Gartner G00173809), 8p.

Burton, B. 2012. *Eight Business Capability Modeling Best Practices Enhance Business and IT Collaboration*. (Gartner G000245455), 17p.

Burton, B. & Allega, P. 2010. *Hype Cycle for Enterprise Architecture, 2010*. (Gartner G00201646), 63p.

Burton, B. & Allega, P. 2011. *Hype Cycle for Enterprise Architecture, 2011*. (Gartner G00214756), 76p.

Business Dictionary. 2010. Web Finance.p. [Online]. Available: <http://www.businessdictionary.com> [Accessed 4 January 2011].

Bygstad, B. & Aanby, H.-P. 2010. ICT Infrastructure for Innovation: a Case Study of the Enterprise Service Bus Approach. *Information Systems Frontiers,* 12(3): 257-265. [Online]. Available: <http://www.springerlink.com/content/0822l76g26m0g712/> [Accessed 11 May 2011].

California Information Technology Council. Enterprise Architecture and Standards Committee. 2005. *California Enterprise Architecture Framework.* 25p. [Online]. Available: <http://www.cio.ca.gov/Publications/pubs/Appendix_A.pdf> [Accessed 7 November 2012].

Cambridge Dictionaries Online. 2011. Cambridge: Cambridge University Press.p. [Online]. Available: <http://dictionary.cambridge.org/dictionary/british/> [Accessed 4 January 2011].

Canada. National Defence. Information Management Group. 2009. *DND/CF Architecture Framework (DNAF) Volume 1: Overview and Definitions*. 19p. [Online]. Available: <http://www.img.forces.gc.ca/pub/af-ca/vol-01/index-eng.asp> [Accessed 4 June 2011].

Cannon, E. 2010. Why Does E-Commerce Need Enterprise Architecture? In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 122-126). Boca Raton: CRC Press.

Carbone, J.A. 2004. *IT Architecture Toolkit*. Upper Saddle River: Prentice Hall PTR. 235p.

CEiSAR. Center for Excellence in Enterprise Architecture. 2008. *Enterprise Modelling: White Paper*. 46p. [Online]. Available:

<http://www.democritique.org/IT/Documents/CEISAR/Enterprise_Modeling_-_CEISAR_-_Apr_08.pdf> [Accessed 3 May 2012].

Chanopas, A., Krairit, D. & Khang, D.B. 2006. Managing Information Technology Infrastructure: a new flexibility framework. *Management Research News,* 29(10): 632-651. [Online]. Available: <http://www.emeraldinsight.com/0140-9174.htm> [Accessed 16 March 2012].

Chase, N. 2006. *Introducing The Open Group Architecture Framework (TOGAF), Part 1: Understand TOGAF and IT architecture in today's world*. 6p. [Online]. Available: <http://www-128.ibm.com/developerworks/ibm/library/ar-togaf1/> [Accessed 23 July 2012].

Chelsom, J.J., Pande, I., Summers, R. & Gaywood, I. 2011. *Ontology-driven Development of a Clinical Research Information System*. Paper presented at the International Symposium on Computer-Based Medical Systems, Bristol. 1-4p. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5999165> [Accessed 7 February 2012].

Cline, D. *s.a. Logical Structure, Theoretical Framework*. [Online]. Available: <http://education.astate.edu/dcline/guide/framework.html> [Accessed 16 August 2012].

Cobban, M. 2004. *What is BPEL and Why is it so Important to my Business?* [Online]. Available: <www.softcare.com/whitepapers/wp_whatis_bpel.php> [Accessed 14 August 2011].

Cook, M.A. 1996. *Building Enterprise Information Architectures: Reengineering Information Systems.* Upper Saddle River: Prentice Hall. 193p.

Creasy, P.N. & Ellis, G.A. 1993. *A Conceptual Graphs Approach to Conceptual Schema Integration*. Paper presented at the International Conference on Conceptual Structures, Quebec, Canada. 16p. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.33.6355> [Accessed 30 April 2012].

Crystal, A. 2007. Facets are Fundamental: Rethinking Information Architecture Frameworks. *Technical Communications,* 54(1): 16-25. [Online]. Available: <http://ils.unc.edu/~acrystal/crystal_facets_are_fundamental_Tech-Comm_2006-final.pdf> [Accessed 4 January 2012].

Damjanovic, V. 2010. Semantic Reengineering of Business Process. *Information Systems,* 35(4): 496-504. [Online]. Available: <http://www.sciencedirect.com/science?_ob=MImg&_imagekey=B6V0G-4WS9BWK-1-1&_cdi=5646&_user=59388&_pii=S0306437909000647&_origin=gateway&_coverDate=06%2F30%2F2010&_sk=999649995&view=c&wchp=dGLzVzb-zSkWA&md5=ee7e3d7669f9423d3bb224fef8a8e2f5&ie=/sdarticle.pdf> [Accessed 7 May 2012].

Davidson, A., Fuchs, M., Hedin, M., Jain, M., Koistinen, J., Lloyd, C. & Maloney, M. 1999, 30 January 2012. *Schema for Object-Oriented XML 2.0*. [Online]. Available: <http://www.w3.org/TR/NOTE-SOX> [Accessed 30 January 2012].

De Vries, M. 2010. A Framework for Understanding and Comparing Enterprise Architecture Models. *Management Dynamics,* 19(2): 17-29. [Online]. Available: <http://search.proquest.com/docview/863259570?accountid=14717> [Accessed 2 May 2012].

De Vries, M. 2012. *A Process Reuse Identification Framework using an Alignment Model.* PhD, University of Pretoria, Pretoria.

DeBoever, L.R., Paras, G.S. & Westbrock, T. 2010. A Pragmatic Approach to a Highly Effective Enterprise Architecture Program. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 156-161). Boca Raton: CRC Press.

Denzin, N.K. & Lincoln, Y.S. (Eds.). 2005. *The SAGE Handbook of Qualitative Research* (3rd ed.). Thousand Oaks: Sage. 1210p.

Dietz, J.L.G., Hoogervost, J.A.P., Albani, A., Aveiro, D., Babkin, E., Barjis, J., Caetoano, A., Huysmans, P., Ijjima, J., Van Kervel, S.J.H., Mulder, H., Op 't Land, M., Proper, H.A., Sanz, J., Terlouw, L., Tribolet, J., Verelst, J. & Winter, R. 2013. The Discipline of Enterprise Engineering. *International Journal of Organizational Design and Engineering,* 3(1): 86-114.

Dodani, M.H. 2008. Aligning IT to Business through Architecture. *Journal of Object Technology,* 7(6): 31-38.

Doig, G. 2007. *Microsoft Services Business Architecture (MSBA): Getting Started with SOA* [Presentation]. [Online]. Available: <http://www.slideserve.com/terrel/microsoft-services-business-architecture-msba-getting-started-with-soa> [Accessed 8 March 2011].

Dols, J. 2008. Business Ownership of Business Architecture: a Case Study. *Cutter IT Journal,* 21(3): 31-37.

Domagalski, Z. 2008. *Information Architecture under the Zachman Framework - Part One: Contextual Modeling*. [Online]. Available: <http://it.toolbox.com/blogs/tipping-point/information-architecture-under-the-zachman-framework-part-one-contextual-modeling-27541> [Accessed 3 February 2012].

Domagalski, Z. 2009. *Information Architecture under the Zachman Framework - Part Two: Conceptual Modeling*. [Online]. Available: <http://it.toolbox.com/blogs/tipping-point/information-architecture-under-the-zachman-framework-part-two-conceptual-modeling-29446> [Accessed 31 January 2012].

Dopfer, K., Foster, J. & Potts, J. 2004. Micro-meso-macro. *Journal of Evolutionary Economics,* 2004(14): 263-279. [Online]. Available: <http://papers.ssrn.com/sol3/papers.cfm?abstract_id=721599> [Accessed 17 November 2012].

Easterbrook, S. 2001. Object Oriented Modeling Methods. 15p. [Online]. Available: <http://www.cs.toronto.edu/~sme/CSC444F/slides/L16-ObjectOrientedModeling.pdf> [Accessed 19 July 2011].

Edmonds, W.A. & Kennedy, T.D. 2013. *An Applied Reference Guide to Research Designs: Quantitative, Qualitative, and Mixed Methods*. Los Angeles: Sage. 213p.

Elzinga, T., Smiers, L. & Van der Vlies, J. 2010. *The CORA Model : a Practical Guide on Using a COmmon Reference Architecture to Design and Deliver Integrated IT Solutions Successfully*. [Online]. Available: <http://www.coramodel.com/> [Accessed 27 March 2012].

The Engineering Design Process. *s.a.* [Online]. Available: <http://www.sciencebuddies.org/engineering-design-process/engineering-design-process-steps.shtml> [Accessed 16 August 2012].

Evernden, R. & Evernden, E. 2003. *Information First: Integrating Knowledge and Information Architecture for Business Advantage*. Oxford: Elsevier. 227p.

Executive Office of the President of the United States. 2007. FEA Consolidated Reference Model Document: Version 2.3. 90p. [Online]. Available: <http://www.cio.gov/Documents/FEA_CRM_v2-3_Oct_2007.pdf> [Accessed 16 June 2011].

Experts Exchange. *s.a.* [Online]. Available: <http://www.experts-exchange.com/Microsoft/Development/MS_Access> [Accessed 19 July 2011].

Few, S. 2006. *Visual Communication: Core Design Principles for Displaying Quantitative Information*. 18p. [Online]. Available: <http://www.perceptualedge.com/articles/Whitepapers/Visual_Communication.pdf> [Accessed 11 September 2012].

Finkelstein, C. 2006. *Enterprise Architecture for Integration: Rapid Delivery Methods and Technologies*. Boston: Artech House. 500p.

Fischer, C., Winter, R. & Aier, S. 2010. What is an Enterprise Architecture Principle?: towards a Consolidated Definition. *Studies in Computational Intelligence,* 317: 193-205. [Online]. Available: <http://web.iwi.unisg.ch/org/iwi/iwi_pub.nsf/wwwPublYearGer/6306E950CF043E06C12

577A6002F1120/$file/Fischer.Winter.Aier.Principles.2010.pdf> [Accessed 15 May 2011].

Ford, J.D. & Mott, R.K. 2007. The Convergence of Technical Communication and Information Architecture: Creating Single-source Objects for Contemporary Media. *Technical Communication,* 54(3): 333-342. [Online]. Available: <http://community.wvu.edu/~bdb026/606/Ford-Mott-Single-Source.pdf> [Accessed 31 January 2012].

Frazier, I. 2010. *7 Design Process Steps Designers Should Follow*. [Online]. Available: <http://inspirationfeed.com/articles/design-articles/design-process-flowchart-that-all-freelancers-designers-must-have/> [Accessed 16 August 2012].

Gall, N. 2012. *Gartner's 2011 Global Enterprise Architecture Survey: EA Frameworks are Still Homemade and Hybrid*. (Gartner G00226400), 16p.

Gangemi, A., Borgo, S., Catenacci, S. & Lehmann, J. 2005. *Task Taxonomies for Knowledge Content: D07*. [Online]. Available: <http://www.loa-cnr.it/Papers/D07_v21a.pdf> [Accessed 13 August 2011].

Garg, A., Kazman, R. & Chen, H.-M. 2006. Interface Descriptions for Enterprise Architecture. *Science of Computer Programming,* 16: 4-15.

Garshol, L.M. 2004. Metadata? Thesauri? Taxonomies? Topic maps! Making sense of it all. *Journal of Information Science,* 30(4): 378-391. [Online]. Available: <http://jis.sagepub.com/content/30/4/378.full.pdf+html> [Accessed 4 April 2012].

Gartner. 2007. *Positioning Enterprise Architecture*. Gartner Enterprise Architecture Foundation Seminar [Presentation].

Gartner. 2008a. *Develop the Business Architecture Viewpoint*. Gartner Enterprise Architecture Foundation Seminar [Presentation].

Gartner. 2008b. *Making EA Actionable*. Gartner Enterprise Architecture Foundation Seminar [Presentation].

Gaver, S. 2012. Why doesn't the FEA Work. (Sept), [Online]. Available: <http://semanticommunity.info/Information_Sharing_Environment/Why_Doesn't_the_FEA_Work> [Accessed 18 December 2014].

Gerber, A., Kotzé, P. & Van der Merwe, A. 2010. *Towards the Formalisation of the TOGAF Content Metamodel Using Ontologies.* Paper presented at the 12th International Conference on Enterprise Information Systems, Madeira, Portugal. [Online]. Available: <http://researchspace.csir.co.za/dspace/bitstream/10204/4075/1/Gerber_2010.pdf> [Accessed 2 May 2012].

Gharajedaghi, J. 2006. *Systems Thinking: Managing Chaos and Complexity : a Platform for Designing Business Architecture* (2nd ed.). Amsterdam: Elsevier. 333p.

Godinez, M., Hechler, E., Koenig, K., Lockwood, S., Oberhofer, M. & Schroeck, M. 2010. *The Art of Enterprise Information Architecture : a Systems Based Approach for Unlocking Business Insight*: IBM Press. 480p. [Online]. Available: <http://proquestcombo.safaribooksonline.com/9780137054947> [Accessed 6 Jan 2012].

Graves, T. 2007. *Rethinking Zachman - a Summary*. Blog. [Online]. <http://weblog.tetradian.com/2007/08/15/zachman-summary/> [Accessed 12 May 2012].

Graves, T. 2010, 11 Aug 2010. *A Question of Who*. Blog. [Online]. <http://weblog.tetradian.com/2010/08/11/a-question-of-who/> [Accessed 12 May 2012].

Gray, D.E. 2009. *Doing Research in the Real World* (2nd ed.). Los Angeles: Sage. 604p.

Gregor, S., Hart, D. & Martin, N. 2007. Enterprise Architecture: enablers of Business Strategy and IS/IT Alignment in Government. *Information Technology & People,* 20(2): 96-120. [Online]. Available: <http://dx.doi.org/10.1108/09593840710758031> [Accessed 3 November 2012].

Guevara, D. & Robertson, B. 2011. *Enterprise Solution Architecture: an Overview*. (Gartner G00219389), 14p.

Haller, T. 2010. What is Information Architecture?: Practical Definitions and Usefully Principles for our Second Decade of Study and Work. *Bulletin of the American Society for Information Science and Technology,* 36(6): 13-15. [Online]. Available: <http://www.asis.org/Bulletin/Aug-10/AugSep10_Haller.pdf> [Accessed 9 January 2012].

Handler, R.A. 2001. Advancing the Business Architecture: a Principle-Centered Approach. *Enterprise Architecture Strategies,* File: 127: 1-4.

Handler, R.A. & Newman, D. 2005. *Make a Compelling Case for Enterprise Information Architecture*. (Gartner G00129962), 6p.

Hasselbring, W. 2000. Information System Integration. *Communications of the ACM,* 43(6): 32-38.

Haverty, M. 2002. Information Architecture without Internal Theory: an Inductive Design Process. *Journal of the American Society for Information Science and Technology,* 53(10): 839-845.

Henderson, J.C. & Venkatrman, N. 1999. Strategic Alignment: Leveraging Information Technology for Transforming Organizations. *IBM Systems Journal,* 39(2/3): 472-484. [Online]. Available: <http://search.proquest.com/docview/222419053/fulltextPDF/136778771455A8C70CD/23?accountid=14717> [Accessed 14 November 2011].

Hert, C.A., Denn, S.O., Gillman, D.W., Oh, J.S., Pattuelli, M.C. & Hernandez, N. 2007. Investigating and Modeling Metadata Use to Support Information Architecture Development in the Statistical Knowledge Network. *Journal of the American Society for Information Science and Technology,* 58(9): 1267-1284. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/asi.20613/pdf> [Accessed 31 January 2012].

Higgins, S. & Hebblethwaite, P. 2006. *GEA White Paper: Information Architecture.* Brisbane: The State of Queensland. 33p. [Online]. Available: <http://www.qgcio.qld.gov.au/SiteCollectionDocuments/Architecture%20and%20Standards/GEA/What_is_Information_Architecture.pdf> [Accessed 4 January 2012].

Hinchcliffe, D. 2009. Fixing Enterprise Architecture: Balancing the Forces of Change in the Modern Organization. *ebiz, Sep*, [Online]. Available: <http://www.ebizq.net/blogs/enterprise/2009/09/fixing_enterprise_architecture.php>

Hofstee, E. 2006. *Constructing a Good Dissertation: a Practical Guide to Finishing a Master's, MBA or PhD on Schedule*. Sandton: EPE. 300p.

Hofstee, W.K.B. 1984. Methodological Decision Rules as Research Policies: a Betting Reconstruction of Empirical Research. *Acta Psychologica,* 56: 93-109. [Online]. Available: <http://www.benwilbrink.nl/publicaties/84hofstee_betting.pdf> [Accessed 19 January 2013].

Hokel, T.A. 2010. The Zachman Enterprise Framework^2: an Overview. 4p. [Online]. Available: <http://www.tefg.com/pdf/EFG-Zachman-Enterprise-Framework2-An-Overview.pdf> [Accessed 23 July 2011].

Holt, J. 2009. Points of view. *ITNOW, 51*(3), 16-18p. [Online]. Available: <http://itnow.oxfordjournals.org/content/51/3/16.full.pdf+html> [Accessed 3 November 2012].

Homann, U. & Tobey, J. 2006. From Capabilities to Services: Moving from a Business Architecture to an IT implementation. 30p. [Online]. Available: <http://msdn.microsoft.com/en-us/library/aa479075.aspx> [Accessed 27 August 2011].

Hoogervost, J. 2004. Enterprise Architecture: Enabling Integration, Agility and Change. *International Journal of Cooperative Information Systems,* 13(3): 213-233.

Hotle, M. & Kyle, A. 2012. *Application Deployment Options through the Pace Layer Lens*. (Gartner G00235351), 7p.

Huhta, S. 2002. IT Architecture - for Small & Mid-Sized Organizations. *The Business Forum Journal, Jan*, [Online]. Available: <http://www.bizforum.org/Journal/www_journalSH002.htm> [Accessed 20 February 2012].

Huysmans, P., Ven, K. & Verelst, J. 2010. *Designing for Innovation: Using Enterprise Ontology Theory to Improve Business-IT Alignment.* Paper presented at the 5th SIKS/BENAIS Conference on Enterprise Information Systems, Eindhoven, The Netherlands. [Online]. Available: <http://www.win.tue.nl/eis2010/lib/exe/fetch.php?media=extendedabstract_19.pdf> [Accessed 8 March 2011].

Hyam, R. 2006. *TOM - The TDWG Ontology Metamodel* [Presentation]: TDWG. [Online]. Available: <http://tdwg2006.tdwg.org/fileadmin/2006meeting/slides/Hyam_TdwgOntologyMetamodel_abs0090.pdf> [Accessed 4 April 2012].

IBM Institute for Business Value. 2005. Component Business Models: Making Specialization Real. 17p. [Online]. Available: <http://www-935.ibm.com/services/us/imc/pdf/g510-6163-component-business-models.pdf> [Accessed 15 June 2012].

IEEE. 2000. *IEEE Std 1471-2000: IEEE Recommended Practice for Architectural Description of Software-Intensive Systems*. 23p. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=875998> [Accessed 30 April 2012].

IFIP–IFAC Task Force on Architectures for Enterprise Integration. 1999. *GERAM: Generalized Reference Architecture and Methodology*. IFIP. 31p.

Instructional Design: Using the ADDIE Model. *s.a.* [Online]. Available: <http://raleighway.com/addie/> [Accessed 16 August 2012].

ISO. 1996. *Open Systems Interconnection - Basic Reference Model: The Basic Model*. *Information Technology,* ISO/IEC 7498-1:1994(E), 68p. [Online]. Available: <http://webstore.iec.ch/preview/info_isoiec7498-1%7Bed2.0%7Den.pdf> [Accessed 4 June 2011].

Iyer, B. & Gottlieb, R. 2004. The Four-Domain Architecture : an Approach to Support Enterprise Architecture Design. *IBM Systems Journal,* 43(3): 587-597.

James, G.A., Handler, R.A., Lapkin, A. & Gall, N. 2005. *Gartner Enterprise Architecture Framework: Evolution 2005*. (Gartner G00130855), 6p.

Jin, M., Kung, D. & Peng, W. 2010. *Research of Information System Technology Architecture.* Paper presented at the Industrial and Information Systems (IIS), 2010 2nd International Conference on Dalian. [Online]. Available: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5565708&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D5565708> [Accessed 20 February 2012].

Johnson, P., Lagerström, R., Närman, P. & Simonsson, M. 2006. *Extended Influence Diagrams for Enterprise Architecture Analysis.* Paper presented at the 10th IEEE International Enterprise Distributed Object Computing Conference. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4031190> [Accessed 31 January 2012].

Jonkers, H., Lankhorst, M.M., Ter Doest, H.W.L., Arbab, F., Bosma, H. & Wieringa, R.J. 2006. Enterprise Architecture: Management Tool and Blueprint for the Organisation. *Information Systems Frontiers,* 8(2): 63-66. [Online]. Available: <http://mmlab.ceid.upatras.gr/courses/AIS_SITE/files/3/Enterprise%20architecture-%20Management%20tool%20and%20blueprint%20for%20the%20organisation.pdf> [Accessed 7 July 2011].

Jupp, V. 2006. *The Sage Dictionary of Social Research Methods*. London: Sage. 335p.

Kaisler, S.H., Armour, F.J. & Valivullah, M. 2005. *Enterprise Architecting: Critical Problems.* Paper presented at the 38th Hawaii International Conference on System Sciences, Hawaii. [Online]. Available: <http://origin-www.computer.org/plugins/dl/pdf/proceedings/hicss/2005/2268/08/22680224b.pdf?template=1&loginState=1&userData=anonymous-IP%253A%253AAddress%253A%2B41.135.163.165%252C%2B%255B140.98.196.192%252C%2B41.135.163.165%252C%2B127.0.0.1%255D> [Accessed 4 June 11].

Kang, D., Lee, J., Choi, S. & Kim, K. 2010. An Ontology-based Enterprise Architecture. *Expert Systems with Applications,* 37(2): 1456-1464. [Online]. Available: <http://www.sciencedirect.com/science?_ob=MImg&_imagekey=B6V03-4WP47MJ-J-R&_cdi=5635&_user=59388&_pii=S0957417409006368&_origin=gateway&_coverDate=03%2F31%2F2010&_sk=999629997&view=c&wchp=dGLbVlz-zSkzk&md5=57fab347e4a243b12e9f530f6bc317ed&ie=/sdarticle.pdf> [Accessed 11 Aug 2011].

Kang, D., Lee, J. & Kim, K. 2010. Alignment of Business Enterprise Architectures using Fact-based Ontologies. *Expert Systems with Applications,* 37(4): 3274-3283. [Online]. Available: <http://www.sciencedirect.com/science?_ob=MImg&_imagekey=B6V03-4X9NV4M-7-W&_cdi=5635&_user=59388&_pii=S0957417409008331&_origin=gateway&_coverDate=04%2F30%2F2010&_sk=999629995&view=c&wchp=dGLzVlz-zSkzV&md5=6c9756fe8cff8f040f5f17fadff41694&ie=/sdarticle.pdf> [Accessed 7 May 2012].

Kappelman, L.A. 2010a. Bridging the Chasm. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 35-36). Boca Rota: CRC Press.

Kappelman, L.A. 2010b. Enterprise Architecture: not just another Management Fad. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 117-121). Boca Raton: CRC Press.

Kappelman, L.A. 2010c. The Pioneers of Enterprise Architecture: a Panel Discussion. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 9-26). Boca Raton: CRC Press.

Kappelman, L.A. (Ed.). 2010d. *The SIM Guide to Enterprise Architecture*. Boca Raton: CRC Press. 284p.

Kappelman, L.A., McGinnis, T., Pettit, A., Salmans, B. & Sidorova, A. 2010. Enterprise Architecture: Charting the Territory for Academic Research. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 96-110). Boca Raton: CRC Press.

Keller, W. 2009. *Using Capabilities in Enterprise Architecture Management*. 16p. [Online]. Available: <http://www.objectarchitects.biz/ResourcesDontDelete/CapabilityBasedEAMWhitepaper.pdf> [Accessed 7 November 2011].

Kim, B.-O. & Lee, S.M. 1996. Logistics Information's Role within an IT Systems Architecture in a World-class Organization. *Logistics Information Management,* 9(3): 19-26.

Kirikova, M. 2009. *Towards Flexible Information Architecture for Fractal Information Systems*. Paper presented at the International Conference on Information, Process, and Knowledge Management, Cancun. 135-140p. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4782578> [Accessed 7 February 2012].

Klouwenberg, M.K., Koo, W.J.D. & Van Schaik, J.A.M. 1995. Establishing Business Strategy with Information Technology. *Information Management & Computer Security,* 3(5): 8-20.

Kluge, C., Dietzsch, A. & Rosemann, M. 2006. *How to Realise Corporate Value from Enterprise Architecture.* Paper presented at the European Conference on Information Systems. [Online]. Available: <http://aisel.aisnet.org/ecis2006/133> [Accessed 4 June 2011].

Kruchten, P. 1995. Architectural Blueprints: The "4+1" View Model of Software Architecture. *IEEE Software,* 12(6): 42-50.

Kruchten, P. 2000. *From Waterfall to Iterative Development : a Challenging Transition for Project Managers*. [Online]. Available: <http://www.ibm.com/developerworks/rational/library/content/RationalEdge/dec00/FromWaterfalltoIterativeDevelopmentDec00.pdf> [Accessed 9 September 2012].

Kumar, A., Raghavan, P., Ramanathan, J. & Ramamth, R. 2008. *Enterprise Interaction Ontology for Change Impact Analysis of Complex Systems*. Paper presented at the IEEE Asia-Pacific Services Computing Conference, Yilan. 303-309p. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4780692> [Accessed 4 April 2012].

Kumar, R. 2011. *Research Methodology: a step-by-step Guide for Beginners* (3rd ed.). Los Angeles: Sage. 415p.

Lankhorst, M.M. 2004. Enterprise Architecture Modelling - the Issue of Integration. *Advanced Engineering Informatics,* 18(4): 205-216. [Online]. Available: <http://ac.els-cdn.com/S1474034605000054/1-s2.0-S1474034605000054-main.pdf?_tid=41cae715ae5d41c94bcb2b732aee5d81&acdnat=1335773298_5f0940b1cad947da07ad3fc8b915e693> [Accessed 7 May 2012].

Lankhorst, M.M., Iacob, M.E., Jonkers, H., Van der Torre, L., Proper, H.A., Arbab, F., De Boer, F., Bonsangue, M. & Hoppenbrouwers, S. 2005. *Enterprise Architecture at Work: Modelling, Communication, and Analysis.* Berlin: Springer. 334p.

Lapkin, A. 2004. *Architecture Frameworks: Some options*. (Gartner G00124384), 8p.

Lapkin, A., Allega, P., Burke, B., Burton, B., Bittler, R.S., Handler, R.A., James, G.A., Robertson, B., Newman, D., Weiss, D., Buchanan, R. & Gall, N. 2008. *Gartner Clarifies the Definition of the Term 'Enterprise Architecture'*. (Gartner G00156559), 5p.

Lin, F. & Dyck, H. 2010. The Value of Implementing Enterprise Architecture in Organizations. *Journal of International Technology and Information Management,* 19(1):

1-18. [Online]. Available: <http://search.proquest.com/docview/840244964/fulltext/12F6E75E1F93E2CB396/2?accountid=14717#> [Accessed 5 February 11].

Lloyd, M. 2006. *Microsoft Motion: Business Architecture Methodology*. Paper presented at the Microsoft Architect Insight Conference. 21p. [Online]. Available: <http://www.google.co.za/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&ved=0CBsQFjAA&url=http%3A%2F%2Fdownload.microsoft.com%2Fdocuments%2Fuk%2Fmsdn%2Farchitecture%2Farchitectinsight%2FDay%25201%2FMethods%2FQAd%2520-%2520Methods%2FMET0102%2520AIC%2520Motion%25206%2520WORDS.ppt&ei=eLHsU7DxPM2e7AaFwYHwCQ&usg=AFQjCNEIQhbSDyn-wlSTxG1ejjPLqTX29A&bvm=bv.72938740,d.ZWU> [Accessed 11 August 2011].

Locke, S. 2010. *Current Framework2 elaboration for Zachman*. [Online]. Available: <http://www.zachmanframeworkassociates.com/index.php/component/content/article/26-articles/100-framework2-for-zachman> [Accessed 5 July 2011].

Long, A.M. 2008. Enterprise Architecture: Origins, Tools and Insights (contemporary issues). *Air Force Journal of Logistics*(32): 4.

Lucchi, R. & Mazzara, M. 2007. A Pi-Calculus Based Semantics for WS-BPEL. *The Journal of Logic and Algebraic Programming,* 70(2007): 96-118.

Macgregor, S. 2008. *Working Towards Realising Boundaryless Information Flow*. Paper presented at the The Open Group - South Africa. 34p. [Online]. Available: <http://opengroup.co.za/sites/default/files/presentations/OG%20Information%20Architecture%2020080415_0.pdf> [Accessed 23 July 2011].

Maree, K. (Ed.). 2007. *First Steps in Research*. Pretoria: Van Schaik. 349p.

Martin, R.A., Walker, L.M. & Robertson, E. 2007. *ISO Standards for Enterprise Architecture: EA Washington DC Metro Chapter 2007 Annual General Meeting* [Presentation]. Washington. [Online]. Available: <http://www.aea-

dc.org/resources/2007-9-6-Richard-Martin-ISO-Standards.pdf> [Accessed 24 November 2012].

McCarthy, I. & Menicou, M. 2002. A Classification Schema of Manufacturing Decisions for the GRAI Enterprise Modelling Technique. *Computers in Industry,* 47(3): 339-355.

McGowan, B. 2007. *Microsoft Motion* [Presentation]. [Online]. Available: <http://docs.google.com/viewer?a=v&q=cache:h85IijG87NgJ:gunston.gmu.edu/ecommerce/mba731/presentation/MBA731%2520Motion%2520presentation.ppt+%22microsoft+motion%22+business+architecture+new&hl=en&gl=za&pid=bl&srcid=ADGEESgV8Cr7oz_-U2p6x-2AtH9kdkIxYlulRK3c96Z6UknO_AOqLfhOP9jT_xCPKwHgVv-yw7C9zNYD9r8c--DkvQ7iyfeayjO7oQJCCShqjRNO56fgnYfFANOItFACAjku9HF8FiMn&sig=AHIEtbSfFfrzV3jy8ZbiIELwy80K43N0oQ> [Accessed 3 Sept 2011].

McGrath, S. 2000. *Pyxie*. [Online]. Available: <http://www.xml.com/pub/a/2000/03/15/feature/index.html> [Accessed 3 February 2012].

McWhorter, N. 2008. Business Architecture and Accountability to Business Intent. *Cutter IT Journal,* 21(3): 11-15.

Merrifield, R. & Tobey, J. 2006. *Motion Lite: A Rapid Application of the Business Architecture Techniques Used by Microsoft Motion*. [Online]. Available: <http://www.microsoft.com/indonesia/msdn/motionlite.aspx> [Accessed 27 August 2011].

Mertz, D. 2002. *XML Matters: Intro to PYX*. *developerWorks.* [Online]. Available: <http://www.ibm.com/developerworks/xml/library/x-matters17/index.html> [Accessed 30 January 2012].

Mingxin, G. 2009. *Ontology Model Developing Based on Semantic Analysis for Inter-view Consistency in Enterprise Architecture.* Paper presented at the IEEE Conference on Grey Systems and Intelligent Services, Nanjing, China. [Online]. Available:

<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5408136> [Accessed 3 November 2012].

Minoli, D. 2008. *Enterprise Architecture A to Z: Frameworks, Business Process Modeling, SOA, and Infrastructure Technology*. Boca Raton: CRC. 481p.

Moon, J. 2000. *SearchStorage*. [Online]. Available: <http://searchstorage.techtarget.com/definition/information-architecture> [Accessed 5 January 2012].

Morar, S. 2008. *One Hour Tutorial to TOGAF for Freshers* [Presentation]: The Open Group. [Online]. Available: <http://www.togaf.biz/TOGAFWebsitefiles/One%20Hour%20Tutorial%20To%20TOGAF.pdf> [Accessed 7 July 2011].

Mouton, J. 2001. *How to Succeed in your Master's and Doctoral Studies*. Hatfield: Van Schaik. 280p.

Mykitshyn, M.G. & Rouse, W.B. 2007. Supporting Strategic Enterprise Processes: an Analysis of various Architectural Frameworks. *Information, Knowledge, Systems Management,* 6(1-2): 145-175. [Online]. Available: <http://web.ebscohost.com/ehost/pdfviewer/pdfviewer?vid=3&hid=122&sid=6abebe9e-195b-4194-8f90-0605dee478f1%40sessionmgr115> [Accessed 30 April 2012].

Neaga, E. & Harding, J. 2005. An Enterprise Modeling and Integration Framework Based on Knowledge Discovery and Data Mining. *International Journal of Production Research,* 43(6): 1089-1108. [Online]. Available: <http://web.ebscohost.com/ehost/pdfviewer/pdfviewer?vid=3&hid=14&sid=1d83b7fa-aa13-4756-8a1d-828533b3d1a5%40sessionmgr15> [Accessed 4 January 2012].

Neef, T. 2011. *Technical Architecture*. 29p. [Online]. Available: <http://www.kino-project.org/documents/Technical%20Architecture%20V4.0.pdf> [Accessed 1 April 2012].

Newman, D., Allega, P., Burton, B., Gall, N., Handler, R.A., Robertson, B., Short, J. & Wilson, C. 2009. *Predicts 2010: Major Changes in Store for Enterprise Architecture*. (Gartner G00172857), 12p.

Newman, D., Gall, N. & Lapkin, A. 2008. *Gartner Defines Enterprise Information Architecture*. (Gartner G00154071), 6p.

Newman, M.W. & Landay, J.A. 2000. *Sitemaps, Storyboards, and Specifications: a Sketch of Web Site Design Practice.* Paper presented at the DIS '00 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques, New York. [Online]. Available: <http://dl.acm.org/citation.cfm?id=347758> [Accessed 7 July 2011].

O'Rourke, C., Fishman, N. & Selkow, W. 2003. *Enterprise Architecture: using the Zachman Framework*. Boston: Thomson. 716p.

Object Management Group. 2008a. *Business Process Modeling Notation, V1.1*. 318p. [Online]. Available: <http://www.omg.org/spec/BPMN/1.1/PDF> [Accessed 11 September 2011].

Object Management Group. 2008b. Semantics of Business Vocabulary and Business Rules (SBVR). 422p. [Online]. Available: <http://www.omg.org/spec/SBVR/1.0/PDF/> [Accessed 31 July 2011].

Object Management Group. 2011. *OMG Meta Object Facility (MOF) Core Specification*. 74p. [Online]. Available: <http://www.omg.org/spec/MOF/2.4.1> [Accessed 7 July 2011].

Object Management Group. 2014. *Unified Modeling Language (UML)*. [Online]. Available: <http://www.omg.org/spec/UML/index.htm> [Accessed 20 April 2014].

Ohren, O.P. 2004. *Ontology for Characterising Architecture Frameworks*. Paper presented at the Conference on Advanced Information Systems Engineering. 368-372p. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.87.9846> [Accessed 4 April 2012].

Okon, W. 2010. *DoD Architecture: Information Sharing Across the Gaps*. Paper presented at the DoD Enterprise Architecture Conference, San Antonio, Texas. 23p. [Online]. Available: <http://www.slideserve.com/amal/dod-architecture-information-sharing-across-the-gaps-prepared-for-afei-inter-agency-and-cross-domain-information-shar#download> [Accessed 9 June 2012].

Open Security Architecture. *s.a. IT Architecture*. [Online]. Available: <http://www.opensecurityarchitecture.org/cms/definitions/it-architecture> [Accessed 20 February 12].

Orr, K. 2003. Frameworks and Processes for Developing Enterprise Architectures. *Enterprise Architecture Advisory Service Executive Update,* 6(2): 1-36.

Orr, K. 2008. Opening Statement. *Cutter IT Journal,* 21(3): 3-5.

Other Architecture and Frameworks. 2006. [Online]. Available: <http://pubs.opengroup.org/architecture/togaf8-doc/arch/chap37.html> [Accessed 27 July 2011].

Oxford Dictionaries. 2011. Oxford University Press.p. [Online]. Available: <http://oxforddictionaries.com/definition> [Accessed 4 January 2011].

Pai, J.-C. & Lee, G.G. 2005. Towards a Strategic Information Architecture (SIA) and its Initial Application. *Journal of Information Science,* 31(2): 149-160.

Papazoglou, M.P. & Van den Heuvel, W.-J. 2007. Service Oriented Architectures: Approaches, Technologies and research Issues. *The VLDB Journal,* 16(3): 389-415. [Online]. Available: <http://delivery.acm.org/10.1145/1270000/1265298/778_2007_Article_44.pdf?ip=137.215.9.20&acc=ACTIVE%20SERVICE&CFID=78945921&CFTOKEN=37435791&__acm__=1327057223_2a62af34331cc326066d60a53f53bb10> [Accessed 20 Jan 2012].

Parrow, J. 2001. An Introduction to the π-Calculus. In J.A. Bergstra *et al.* (Eds.), Handbook of Process Algebra (1st ed., pp. 479-543). Amsterdam: Elsevier. [Online].

Available: <http://www.sciencedirect.com/science/book/9780444828309> [Accessed 11 August 2011].

Pascot, D., Bouslama, F. & Mellouli, S. 2011. Architecturing Large Integrated Complex Information Systems: an Application to Healthcare. *Knowledge and Information Systems,* 2011(27): 115-140. [Online]. Available: <http://www.springerlink.com/content/d431410r43350812/fulltext.pdf> [Accessed 8 March 2011].

Perks, C. & Beveridge, T. 2004. Introduction to Technical ArchitectureGuide to Enterprise IT Architecture (pp. 1-20). New York: Springer Verlag. [Online]. Available: <http://books.google.co.za/books?id=LT384bEg_ssC&pg=PA1&source=gbs_toc_r&cad=3#v=onepage&q&f=false> [Accessed 28 February 2012].

Peslak, A.R. 2008. Organizational Information Systems: Rate of Return and Influencing Variables as Viewed by Top Financial Executives. *Industrial Management & Data Systems,* 108(1): 43-59. [Online]. Available: <http://www.emeraldinsight.com/journals.htm?articleid=1657730&show=html> [Accessed 5 February 2011].

Pessi, K., Magoulas, T. & Hugoson, M.-A. 2011. Enterprise Architecture Principles and their Impact on the Management of IT Investments. *The Electronic Journal Information Systems Evaluation,* 14(1): 53-62.

Quin, L. 2012. *Extensible Markup Language (XML)*. [Online]. Available: <http://www.w3.org/XML/> [Accessed 31 January 2012].

Radhakrishnan, R. 2006. USA Patent No. US 7,143,420 B2. USA Patent Office. 25p. [Online]. Available: <http://www.google.com/patents/US7143420> [Accessed 6 January 2011].

Ramakrishnan, R. *s.a.* The Relational Model : Module 1, Lecture 2. 19p. [Online]. Available:

<http://pages.cs.wisc.edu/~dbbook/openAccess/firstEdition/slides/pdfslides/mod1l2.pdf> [Accessed 19 July 2011].

Rasool, F. 2011. Africa's CIO name 2011's big five. *ITWeb Business*, [Online]. Available: <http://www.itweb.co.za/index.php?option=com_content&view=article&id=40791&catid=69> [Accessed 7 February 2011].

Rathwell, G. *s.a. CIMOSA : European Enterprise Integration Concept*. [Online]. Available: <http://www.pera.net/Arc_cimosa.html> [Accessed 5 February 2011].

Rehan, M. & Akyuz, G.A. 2010. Enterprise Application Integration (EAI), Service Oriented Architectures (SOA) and their Relevance to e-supply Chain Formation. *African Journal of Business Management,* 4(13): 2604-2614. [Online]. Available: <http://www.academicjournals.org/ajbm/pdf/pdf2010/4Oct/Rehan%20and%20Akyuz.pdf> [Accessed 4 January 2012].

Ricca, M. 2011. *Bridging the Gap between Enterprise Architecture and Solution Architecture for Maximum Benefit*. [Online]. Available: <http://www.oracle.com/technetwork/articles/entarch/ricca-bridging-ea-sa-457138.html> [Accessed 14 November 2011].

Rico, D.F. 2006. A Framework for Measuring ROI of Enterprise Architecture. *Journal of Organizational Computing and End User Computing,* 18(2): i-xii. [Online]. Available: <http://search.proquest.com/docview/199922613/fulltextPDF?accountid=14717> [Accessed 5 February 2011].

Roberts, J.P. 2002. *What Makes IT Enterprise Architecture Successful*. (Gartner COM-16-8553), 3p.

Robertson, B. 2005. *A Framework of Patterns, Services, Domains and Components Defines the Technology Viewpoint*. (Gartner G00135179), 20p.

Robertson, B. 2006. *EA Modeling Made Simple: ETA Domain Strategy Example*. (Gartner G00144602), 10p.

Robertson, B. 2010. *IT Architecture is not Enterprise Architecture*. (Gartner G00206910), 7p.

Roets, G.J. 2012. *Paper Model of the REAM*. [Unpublished]

Rohloff, M. 2008, 16-20 March 2008. *An Integrated View on Business - and IT-architecture.* Paper presented at the ACM Symposium on Applied Computing, Ceara, Brazil. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1363822> [Accessed 15 Nov 2012].

Rosen, M. 2008. The Business Motivation Model: Matching the Means to the Ends. *Cutter IT Journal,* 21(3): 6-10.

Rosen, M. 2010. Enterprise Architecture 2010: Part 1 - Current Practices. *Enterprise Architecture Advisory Service Executive Update,* 13(7): 1-6.

Rosenfeld, L. & Morville, P. 2002. *Information Architecture: for the World Wide Web* (2nd ed.). Sebastopol: O'Reilly. 496p. [Online]. Available: <http://shop.oreilly.com/product/9780596000356.do> [Accessed 9 January 2012].

Rosenfeld, L. & Morville, P. 2007. *Information Architecture: for the World Wide Web* (3rd ed.). Sebastopol: O'Reilly. 504p. [Online]. Available: <http://books.google.co.za/books?id=OM3DvakML-Mc&printsec=frontcover#v=onepage&q&f=false> [Accessed 9 January 12].

Ross, J.R., Weill, P. & Robertson, D.C. 2006. *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution*. Boston: Harvard Business School Press. 234p.

Rosser, B. 2002. *A Practical Format for IT Architecture Guidelines*. (Gartner TU-14-4319), 5p.

Rummler, G.A. & Brache, A.P. 1995. *Improving Performance: How to Manage the White Space on the Organization Chart* (2nd ed.). San Francisco: Jossey-Bass. 226p.

Saat, J., Franke, U., Lagerström, R. & Ekstedt, M. 2010. *Enterprise Architecture Models for IT/Business Alignment Situations.* Paper presented at the 14th IEEE International Enterprise Distributed Object Computing Conference. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5630223> [Accessed 30 Apr 2012].

Sack, W. & Hearst, M. 2001. *Information Architecture and Web Site Design* [Slides]. Berkeley: University of California. [Online]. Available: <http://courses.ischool.berkeley.edu/i202/f02/Lectures//Lecture26_2002_12_03_tbd.ppt> [Accessed 9 Jan 2012].

Saha, P. 2007. A Synergistic Assessment of the Federal Enterprise Architecture Framework against GERAM (ISO 157404:2000). In P. Saha (Ed.), *Handbook of Enterprise Systems Architecture in Practice* (pp. 1-17): Idea Group.

Salmans, B. 2010. EA Maturity Models. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 89-95). Boca Raton: CRC Press.

Salmans, B. & Kappelman, L.A. 2010. The State of EA: Progress, not Perfection. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 165-217). Boca Raton: CRC Press.

Schekkerman, J. 2004a. *A Comparative Survey of Enterprise Architecture Frameworks* [Presentation]: Institute for Enterprise Architecture Developments. [Online]. Available: <http://www.google.co.za/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&ved=0CC8QFjAA&url=http%3A%2F%2Fwww.enterprise-architecture.info%2FImages%2FDocuments%2FComparative_Survey_of_EA_Frameworks.pps&ei=ms7sU-n2G4GM7QaMhYG4Ag&usg=AFQjCNHwfMPKPGj3GHeHK5Wqao8xwe3Pgg> [Accessed 7 July 2011].

Schekkerman, J. 2004b. *How to survive in the Jungle of Enterprise Architecture Frameworks: creating or choosing an Enterprise Architecture Framework.* (2nd ed.). Victoria: Trafford. 222p.

Schelp, J. & Stutz, M. 2007. A Balanced Scorecard Approach to Measure the Value of Enterprise Architecture. *Journal of Enterprise Architecture,* 3(4): 5-11. [Online]. Available: <http://www.mendeley.com/research/a-balanced-scorecard-approach-to-measure-the-value-of-enterprise-architecture/#> [Accessed 4 June 2011].

Schlier, F.W. 2006. *Achieving Agility: Information Architecture is a Key Component of Enterprise Agility*. (Gartner G00138135), 6p.

Schultz, M. 2007. *Architecture Principles: Creating the Foundation for Robust Architecture*. *IBM developerWorks.* [Online]. Available: <http://www.ibm.com/developerworks/library/ar-archprinc/> [Accessed 20 October 2012].

Sehmi, A. 2008. *UPDATE - Business Architecture Resources (MSBA / Motion)* [Online]. <http://blogs.msdn.com/b/asehmi/archive/2008/03/03/update-business-architecture-resources-msba-motion.aspx> [Accessed 20 June 2014].

Sessions, R. 2007. A Comparison of the top four Enterprise-Architecture Methodologies. 36p. [Online]. Available: <http://msdn.microsoft.com/en-us/library/bb466232.aspx#> [Accessed 7 July 2011].

Sessions, R. 2008. *An Introduction to the Major EA Methodologies* [Presentation]: ObjectWatch. [Online]. Available: <http://www.agileea.com/Whitepapers/2008-05-30-RogerSessions-EA.pdf> [Accessed 7 July 2011].

Shah, H. & El Kourdi, M. 2007. Frameworks for Enterprise Architecture. *IT Professional Magazine,* 9(5): 36-41. [Online]. Available: <http://search.proquest.com/docview/206325193> [Accessed 16 March 2012].

Shi, S. & Zhang, R. 2009. *SXML, an Enhancement of XML Documents in Mobile Learning.* Paper presented at the International Conference on Computational Intelligence and Software Engineering, 2009, Wuhan. [Online]. Available: <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5362995&tag=1> [Accessed 30 January 2012].

Sidorova, A. & Kappelman, L.A. 2010. Enterprise Architecture as Politics: an Actor-Network Theory Perspective. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture.* (pp. 70-88). Boca Raton: CRC Press.

Simons, G.F., Kappelman, L.A. & Zachman, J.A. 2010. Using Language to Gain Control of Enterprise Architecture: on the verge of major Business Reengineering. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 127-146). Boca Raton: CRC Press.

Singh, S. 2011. *Enterprise Architecture: an Overview*. 175p. [Online]. Available: <http://www.scribd.com/doc/155097694/Enterprise-Architecture-an-overview> [Accessed 11 January 2012].

Slay, J. 2002. *A Cultural Framework for the Interoperability of C2 Systems*. Paper presented at the ICCRTS2002. 12p. [Online]. Available: <http://www.dodccrp.org/events/2002_CCRTS/Tracks/pdf/072.PDF> [Accessed 5 February 2011].

Smit, M. 2008. *The Cyclical Iterative Design Process and Design Environment: Learning from Experience.* PhD, Technische Universiteit Delft, Delft. [Online]. Available: <http://bk.tudelft.nl/fileadmin/Faculteit/BK/Onderzoek/Graduate_School/Thesis/Summary-PhD-Thesis-ENG.pdf> [Accessed 9 September 2012].

SOA Consortium EA2010. Working Group. 2010. Business Architecture: The Missing Link between Business Strategy and Enterprise Architecture. *BPTrends,* February: 1-12. [Online]. Available: <http://www.bptrends.com> [Accessed 20 Feb 12].

Sousa, P., Lima, J., Sampaio, A. & Pereira, C. 2009. *An Approach for Creating and Managing Enterprise Blueprints: a Case for IT Blueprints.* Paper presented at the CAiSE 2009, Amsterdam. [Online]. Available: <http://books.google.co.za/books?hl=en&lr=&id=P6sAYDPcl6gC&oi=fnd&pg=PA70&ots=oY34g_KQrS&sig=krvcqocStUyKOGk1q0tVXR5buxo&redir_esc=y#v=onepage&q&f=false> [Accessed 28 February 2012].

State of North Carolina. Office of the Chief Information Officer. 2005. North Carolina Statewide Technical Architecture: Application Domain. 25p. [Online]. Available: <http://www.ncsta.gov/library/pdf/Application.pdf> [Accessed 2 April 2012].

Steen, M.W.A., Akehurst, D.H., Ter Doest, H.W.L. & Lankhorst, M.M. 2004. *Supporting Viewpoint-Oriented Enterprise Architecture*. Paper presented at the Enterprise Distributed Object Computing Conference, Monterey, California. 201-211p. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1342516> [Accessed 30 April 2012].

Steen, M.W.A., Strating, P., Lankhorst, M.M., Ter Doest, H.W.L. & Iacob, M.-E. 2005. Service-Oriented Enterprise Architecture. In Z. Stojanovic & A. Dahanayake (Eds.), *Service-Oriented Software System Engineering: Challenges and Practices* (pp. 132-154). Hershey: Idea Group.

Stevenson, D.A. 2003. Enterprise Architecture: Business Themes and Enterprise Architecture. [Online]. Available: <http://users.iafrica.com/o/om/omisditd/denniss/text/busthem0.html> [Accessed 4 June 2011].

Swanborn, P.G. 2010. *Case Study Research: What, Why and How?* Los Angeles: Sage. 178p.

Sweeney, S. 2002. *SLiP and SLIDE : a quick XML shorthand syntax and tool for editing*. [Online]. Available: <http://mail.python.org/pipermail/xml-sig/2002-April/007538.jtml> [Accessed 30 January 2012].

Tang, A., Han, J. & Chen, P. 2004. *A Comparative Analysis of Architecture Frameworks*. (S.U.o. Technology SUTIT-TR2004.01), 15p. [Online]. Available: <http://www.swin.edu.au/ict/research/technicalreports/2004/SUTIT-TR2004.01.pdf> [Accessed 17 October 2012].

Tash, J. 2006. *Enterprise Architecture vs. Technology Architecture*. [Online]. Available: <http://www.flashmapsystems.com/resources/articles/articles29.htm> [Accessed 20 February 2012].

Teeuwen, P., Van den Berg, M., Van Alen, R. & Luijpers, J. 2010. Project Start Architecture Part I : An Effective Way to Translate EA into Projects. *Enterprise Architecture Advisory Service. Executive Update,* 13(5): 1-4.

Thario, J. 2008. *Applicability of DoDAF in Documenting Business Enterprise Architectures*. Blog. [Online]. <http://www.thario.net/2008/08/applicability-of-dodaf-in-documenting.html> [Accessed 9 July 2011].

The Information Architecture Institute. 2007. *About Us*. [Online]. Available: <http://iainstitute.org/en/about/our_mission.php> [Accessed 5 January 2012].

The Open Group. 2004. *Business Executive's Guide to IT Architecture : Why IT Architecture is Crucial to the Success of your Business*. The Open Group. 8p. [Online]. Available: <http://www.immagic.com/eLibrary/ARCHIVES/GENERAL/OPGRP_US/O040920B.pdf> [Accessed 20 February 2012].

The Open Group. 2009a. *TOGAF : Version 9 Enterprise Edition*. [Online]. Available: <http://pubs.opengroup.org/architecture/togaf9-doc/arch/> [Accessed 7 July 2011].

The Open Group. 2009b. *TOGAF Online: Part IV : Architecture Content Framework*. [Online]. Available: <http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap35.html> [Accessed 14 November 2011].

The Open Group. 2009c. *TOGAF Version 9: The Open Group Architecture Framework (TOGAF)*. 778p. [Online]. Available: <http://www.opengroup.org/architecture/togaf91/downloads.htm#Non-Member> [Accessed 4 June 2011].

Thomas, G. 2011. *How to do Your Case Study*. Los Angeles: Sage. 231p.

Thorn, S. 2007. TOGAF and ITIL: a White Paper. 26p. [Online]. Available: <http://www2.opengroup.org/ogsys/jsp/publications/PublicationDetails.jsp?publicationid=12110> [Accessed 3 February 2011].

Thundatil, N. 2007. Architecture for Implementing a Sustainable Electronic Delivery Services Platform: at mitchell.com case study. *The Electronic Library,* 25(4): 465-477. [Online]. Available: <www.emeraldinsight.com/0264-0473.htm> [Accessed 16 March 2012].

Toms, E.G. 2002. Information Interaction: Providing a Framework for Information Architecture. *Journal of the American Society for Information Science and Technology,* 53(10): 855-862. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/asi.10094/pdf> [Accessed 31 Jan 2012].

University of California. 2000. *UC2010: A New Business Architecture for the University of California*. Paper presented at the CUMREC 2002. 64p. [Online]. Available: <http://uc2010.ucsd.edu> [Accessed 11 August 2011].

Urbaczewski, L. & Mrdalj, S. 2006. A Comparison of Enterprise Architecture Frameworks. *Issues in Information Systems,* VII(2): 18-23.

USA. Department of Commerce. Enterprise IT Architecture Advisory Group. 2004. Information Technology Architecture: what is it, why should you care, and how do you do one? , 1-9p. [Online]. Available: <http://ocio.os.doc.gov/s/groups/public/@doc/@os/@ocio/@oitpp/documents/content/prod01-002345.pdf> [Accessed 6 January 2011].

USA. Department of Defence. 2005. *The State of DOD Architecting*. Paper presented at the Command Information Superiority Architectures (CISA), Omaha. 44p. [Online]. Available: <http://www.google.co.za/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&ved=0CBsQFjAA&url=http%3A%2F%2Fwww.modaf.com%2Ffile_download%2F17&ei=t9zsU-mYAuiR7Aaq84HYBw&usg=AFQjCNEXLKheDjKpNZiscXlxGb0-nnZdBA&bvm=bv.72938740,d.ZGU> [Accessed 5 February 2011].

USA. Department of Defence. 2007a. *DoD Architecture Framework Version 1.5: Volume I: Definitions and Guidelines.* 46p. [Online]. Available: <http://dodcio.defense.gov/Portals/0/Documents/DODAF/DoDAF_Volume_I.pdf> [Accessed 5 February 2011].

USA. Department of Defence. 2007b. *DoD Architecture Framework Version 1.5: Volume II: Product Descriptions.* 284p. [Online]. Available: <http://dodcio.defense.gov/Portals/0/Documents/DODAF/DoDAF_Volume_II.pdf> [Accessed 5 February 2011].

USA. FEA Program Management Office. 2004. *Federal Enterprise Architecture* [Presentation]. [Online]. Available: <http://www.whitehouse.gov/sites/default/files/omb/assets/omb/egov/documents/FEA_Overview.pdf> [Accessed 5 February 2011].

USA. Federal Chief Information Officers Council. 2010. *Federal Enterprise Architecture Security and Privacy Profile: Version 3.0.* 85p. [Online]. Available: <http://www.cio.gov/documents/fea-security-privacy-profile-v3-09-30-2010.pdf> [Accessed 16 June 2011].

USA.Chief Information Officer Council. 2001. *A Practical Guide to Federal Enterprise Architecture: v1.0.* 112p. [Online]. Available: <http://www.gao.gov/assets/590/588407.pdf> [Accessed 12 Jun 2012].

Vail, E.F., III. 2002. Causal Architecture: Bringing the Zachman Framework to Life. *Information Systems Management,* 19(3): 8-19.

Veasey, P.W. 2001. Use of Enterprise Architectures in Managing Strategic Change. *Business Process Management Journal,* 7(5): 420-436.

Verbert, K., Klerkx, J., Meire, M., Najjar, J. & Duval, E. 2004. Towards a Global Component Architecture for Learning Objects: an Ontology Based Approach. *Lecture Notes in Computer Science,* 3292: 713-722. [Online]. Available:

<http://lirias.kuleuven.be/bitstream/123456789/125459/1/558_11340553_Chapter_82.pdf> [Accessed 31 January 2012].

Vernadat, F.B. 2007. Interoperable Enterprise Systems: Principles, Concepts, and Networks. *Annual Reviews in Control,* 31(1): 137-145. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1367578807000132> [Accessed 7 May 2012].

Versteeg, G. & Bouwman, H. 2006. Business Architecture: a new Paradigm to Relate Business Strategy to ICT. *Information Systems Frontiers,* 8(2): 91-102. [Online]. Available: <http://www.springerlink.com/content/6w7180576x4x71t7/> [Accessed 7 May 2012].

Washington State. Department of Transport. 2001. *Technology Architecture Plan Template*. Washington. 33p. [Online]. Available: <http://www.google.co.za/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&ved=0CBsQFjAA&url=http%3A%2F%2Fwww.ewita.com%2Fnewsletters%2F10022Files%2FWSDOTArchPlanTemplate.doc&ei=IuDsU7mgDcbD7AaZ6IGQDQ&usg=AFQjCNGSjRNkoK62Hc8V1OX1cyuwO2uy8A> [Accessed 5 February 2011].

Webster's New World Dictionary. 2011. Love to Know Corp.p. [Online]. Available: <http://www.yourdictionary.com> [Accessed 4 January 2011].

Wegmann, A. 2003. *On the Systemic Enterprise Architecture Methodology [SEAM]*. Paper presented at the International Conference on Enterprise Information Systems. 483-490p. [Online]. Available: <http://infoscience.epfl.ch/record/89690/files/Wegmann_ICEIS2003.pdf> [Accessed 7 June 2011].

Westbrock, T. 2007. *Do Frameworks Really Matter?* [Presentation]: EA Directions. [Online]. Available: <http://pdf.aminer.org/000/226/026/improving_eas_for_sequencing_problems.pdf> [Accessed 7 July 2011].

WhatIs.com. 2012. TechTarget.p. [Online]. Available: <http://whatis.techtarget.com/search/query?q=integration> [Accessed 29 May 2012].

Whittle, R. & Myrick, C.B. 2005. *Enterprise Business Architecture: the Formal Link between Strategy and Results*. Boca Raton: Auerbach Publications. 229p.

Winter, R. & Fischer, R. 2007. Essential Layers, Artifacts, and Dependencies of Enterprise Architecture. *Journal of Enterprise Architecture,* May: 1-12. [Online]. Available: <http://www.alexandria.unisg.ch/export/DL/Robert_Winter/204958.pdf> [Accessed 24 Nov 2011].

Wiseman, P. 2007. *Information Technology's Role in Enterprise Architecture* [Presentation]: State of Montana, Information Technology Services Division. [Online]. Available: <http://pdf.aminer.org/000/226/026/improving_eas_for_sequencing_problems.pdf> [Accessed 6 January 2011].

Wu, J.C.-Z. 2007. Enterprise Architecture: a Cross-cutting Solution. *Cutter Enterprise Architecture Advisory Service Executive Update,* 10(6): 1-4.

Wurman, R.S. & Bradford, P. 1996. *Information Architects*. Zurich: Graphis. 235p.

Yee, K.-P., Swearingen, K., Li, K. & Hearst, M. 2003, 2003. *Faceted Metadata for Image Search and Browsing.* Paper presented at the 2003 Conference on Human Factors in Computing Systems (CHI'03), Fort Lauderdale. [Online]. Available: <http://kevinli.net/flamenco-chi03.pdf> [Accessed 4 June 2012].

Yin, R.K. 2009. *Case Study Research: Design and Methods* (4th ed.). Los Angeles: Sage. 219p.

Yin, R.K. 2011. *Qualitative Research form Start to Finish*. New York: Guilford Press. 348p.

Yin, R.K. 2012. *Applications of Case Study Research* (3rd ed.). Los Angeles: Sage. 231p.

Zachman, J.A. 1996. Concepts of the Framework for Enterprise Architecture: Background, Description and Utility. 7p. [Online]. Available: <http://links.enterprisearchitecture.dk/links/files/Zachman_ConceptsforFrameworkforEA.pdf> [Accessed 2 May 2011].

Zachman, J.A. 2008. *John Zachman's Concise Definition of the Enterprise Framework*. 1p. [Online]. Available: <http://old.zachmaninternational.com/concise%20definition.pdf> [Accessed 4 June 2011].

Zachman, J.A. 2010. Architecture is Architecture is Architecture. In L.A. Kappelman (Ed.), *The SIM Guide to Enterprise Architecture* (pp. 37-45). Boca Raton: CRC Press.

Zachman, J.A. 2011. *The Zachman Framework for Enterprise Architecture : The Enterprise Ontology* [Figure]. (Version 3.0).

Zaïane, O.R. 1995. *The Object Oriented Model*. [Online]. Available: <http://www.cs.sfu.ca/CC/354/zaiane/material/notes/Chapter1/node8.html> [Accessed 19 July 2011].