A Reference Model for Trust Framework based on Identity Ecosystem

Amit Gurung, 43170048

A thesis submitted to Macquarie University for the degree of
Masters of Research
Department of Computing
Faculty of Science and Engineering

April 2015

Declaration

I certify that the work in this thesis entitled "*A Reference Model for Trust Framework based on Identity Ecosystem*" has not previously been submitted for a degree nor has it been submitted as part of the requirements for a degree to any other university or institution other than Macquarie University. I also certify that the thesis is an original piece of research and it has been written by me. Any help and assistance that I have received in my research work and the preparation of the thesis itself have been appropriately acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

_____

Amit Gurung

Acknowledgements

I would like to thank my supervisor Dr. Stephen Smith for the guidance, expertise, resources and encouragement during the preparation of this thesis.

Abstract

This research work suggests a suitable reference model to develop Trust Frameworks required in Identity Ecosystem proposed by "*National Strategy for Trusted Identities in Cyberspace*". Trust Frameworks in Identity ecosystem allows communication with many other frameworks, external users, and departments of private or government organizations. The proper sharing of the digital identity in secure way is the core functionality of this ecosystem. Trust Frameworks need to be governed by policies, rules and regulations from the different views, perspectives and needs to have capacity of being agile, interoperable, flexible and secured. The reference model for Trust Framework must have agile behaviour to adapt the new requirements and changes in models, views, methods, processes and perspectives. Various process models are created to maintain the synchronization between the programs created by different assumptions to build the complex systems. Enterprise architecture frameworks are used to manage these complex systems.

Technically, the research compares enterprise architecture frameworks in first phase; secondly it elaborates the finding with an adaptive system development model. The research classifies enterprise architecture frameworks from different enterprise methods/processes, domains, roles, artefacts, tools, and perspectives to evaluate their strength in agility, interoperability, flexibility and security layers. The previous works, comparisons from different perspectives are elaborated here as a proof for the need of different development process like Gill Framework that has adaptability characteristics to build the reference model for Trust Frameworks.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Background

Throughout the history of Internet, online systems are the most important platform for business innovations and changing the human life to the better advancement. This venue provides digital infrastructure for everyone to create identities in digital format. Cyberspace is an environment to keep individual information secure and safe with the encryption of unique personal identity and password. The current environment has millions of usernames with passwords for each user in each application. This interaction is the most typical complication seen in user experience where users have to reuse their access credentials. This kind of behaviour had and may lead the failure of the systems security as the information reused can be a guess, identities can be stolen or passwords may be cracked using various decryption algorithms. And this results the loss of trust on online systems. The authentication factor of any users has been always the critical issues for private and government sectors. As the complexity of information tracking is getting expensive, and requires high security, the current online systems have to be managed in effective way. The solution could be the national strategy for trusted identities in cyberspaces to minimize threats and vulnerable transactions.

After US government published "*National Strategy for Trusted Identities in Cyberspace (NSTIC)*" [36], it has been a top concern for everyone, as the existing digital infrastructure may not satisfy its operational needs based on current situation of Internet. This created a lot of news in media and became a highly debated topic in public for its implementation. It was about the security or privacy; many people find it as just another way to control people's right and privacy. Some find this policy allows government to keep their eye on their people's movement, which could violate the law of privacy act. In general, it is still a new thing for every country to implement this strategy. And there are still only few documents written to review the proposed idea.

The "*National Strategy for Trusted Identities in Cyberspace (NSTIC)*" is the strategy between public and private sectors to create higher level of trust for the identities of individuals, organizations, networks, services and devices, involved in the online systems and interactions. This strategy is the base for easy use and interoperable identities for individuals, organizations and systems that take part in online transactions in the manner of promoting confidence and privacy. This strategy is the base for the development of the user-centric "Identity Ecosystem" environment where each participant will be responsible for the trustworthy transactions and agree for the standard and authenticated digital identities [36]. This Identity Ecosystem is designed to secure the transactions with following basic policies of security. The ecosystem is based on the principles of privacy protection, convenience, efficiency, ease-of-use, security, confidence, innovation and choice. For example, the ecosystem will allow minimum necessary information required for transactions without sharing all the identities of a user. The strategy will be having successful execution only after the proper implementation of all the principle guidelines. Both private and government organizations should align and agree for all development and implementation phases of this Identity Ecosystem. This agreement strategy should be market driven, innovative and adaptive to changes. Usually, the federal government should be responsible for the strategy of the Identity Ecosystems and private sector should lead the development and implementation of this Identity Ecosystem [36].

1.2 Context of the study

The strategy proposed "Identity Ecosystem", as a solution for creating unique digital identities to share among the systems to identify the users. This Identity Ecosystem is the integrated environment of various online communities that use interoperable technology, processes, and policies. It takes long time to develop but always has baseline of privacy, interoperability, and security [36]. This process includes the core part as "Trusted Frameworks" to communicate with different systems, users and third parties. The process to build these Trusted Frameworks could be possible with any enterprise architecture frameworks. As they are the specific methods used to build any complex system. But from the Chapter 2 below, it shows that it is literally

difficult to select one of them without comparing their methods/processes, domains, roles, artefacts, tools and other different attributes. Another problem is the system development process, which needs to be agile enough to adapt to the changes in the enterprise requirements.

The context of the study is to find the possible enterprise Architecture Framework that can be referenced and solution for the agile development process.

The core objective of the research is ***to suggest a relevant reference model to build Trusted Frameworks*** for the Identity Ecosystem.

1.3 Objective of the study

The research started with the topic "*National Strategy for Trusted Identities in Cyberspace*", which was interesting and novel idea for the future digital world. The proposal was strong and specific to the solution for enhancing online choice, efficiency, adequate security and privacy protection. The proposal had the idea of Trust Framework in the Identity Ecosystem, which is built around defined roles/responsibilities, models, policies, process and standards. The purpose was to suggest reference model, which is adaptive in nature, so that it can be used to build other Trust Frameworks. There comes the enterprise architecture framework as a solution to provide the guidelines for building an enterprise system. The research requires the comparison of existing enterprise architecture frameworks and selects the most relevant method as a reference model to build Trust Frameworks.

Adaptive or agility will be most important perspectives for evaluating each enterprise architecture frameworks. The research analysis requires reviewing the current status of the major enterprise architecture frameworks that is agile and interoperable. The model should offer the concept of service science, and system thinking concepts from every aspect. Comparison of enterprise architecture frameworks needs the analysis of methods, processes, domains, roles, artefacts,

perspectives, abstraction and tools offered by Frameworks. This research has two main objectives:

- Evaluation of existing enterprise architecture frameworks to find if any of them can be used directly as reference model to build Trust Frameworks.
- Reference Model should be adaptive in nature to cover the future enterprise changes.

The purposed adaptive reference model should have following eight elements of system thinking concepts: System, Autonomous, Interdependent, Integrated, Context Aware, Adaptive, Self-organizing, and Lifecycle [31].

1.4 National Strategy for Trusted Identities in Cyberspace

The vision of "National Strategy for Trusted Identities in Cyberspace" states that individuals and organizations can utilize secure, efficient, easy-to-use, and interoperable identity solutions to access online services in a manner that promotes confidence, privacy, choice, and innovation [36].

This idea for Identity Ecosystem was based on building a comprehensive framework that can be implemented to enhance the confidence in the use of online systems ensuring the long-term success. Identity Ecosystem was based on the most recommended four Guiding Principles:

- Identity solutions will be privacy-enhancing and voluntary
- Identity solutions will be secure and resilient
- Identity solutions will be interoperable
- Identity solutions will be cost-effective and easy to use

These principles were the foundation for all of the Strategy's goals, objectives, and actions. The success of strategy and ideal fulfilled of the Identity Ecosystem are completely dependent on above Guiding Principles. The Identity Ecosystem consists of different online communities that use interoperable technology, processes, and

policies. These communities are managed by Trust Frameworks. The need of the proper Trust Frameworks is the baseline of this research.

1.5 Identity Ecosystem

The above strategy have proposed "Identity Ecosystem" as shown in figure 1.1, as a solution for creating unique digital identities to share among the systems to identify the users. This strategy consists of participants, policies, processes, and technologies required to share the trusted identifications, authentication and authorization in any online transactions over the Internet. Identity Ecosystem has clearly defined its policy components to build the secure online environment. These components are as:

- Identity Ecosystem framework
- Steering group
- Trust framework
- Accreditation authority
- Trustmark schemes

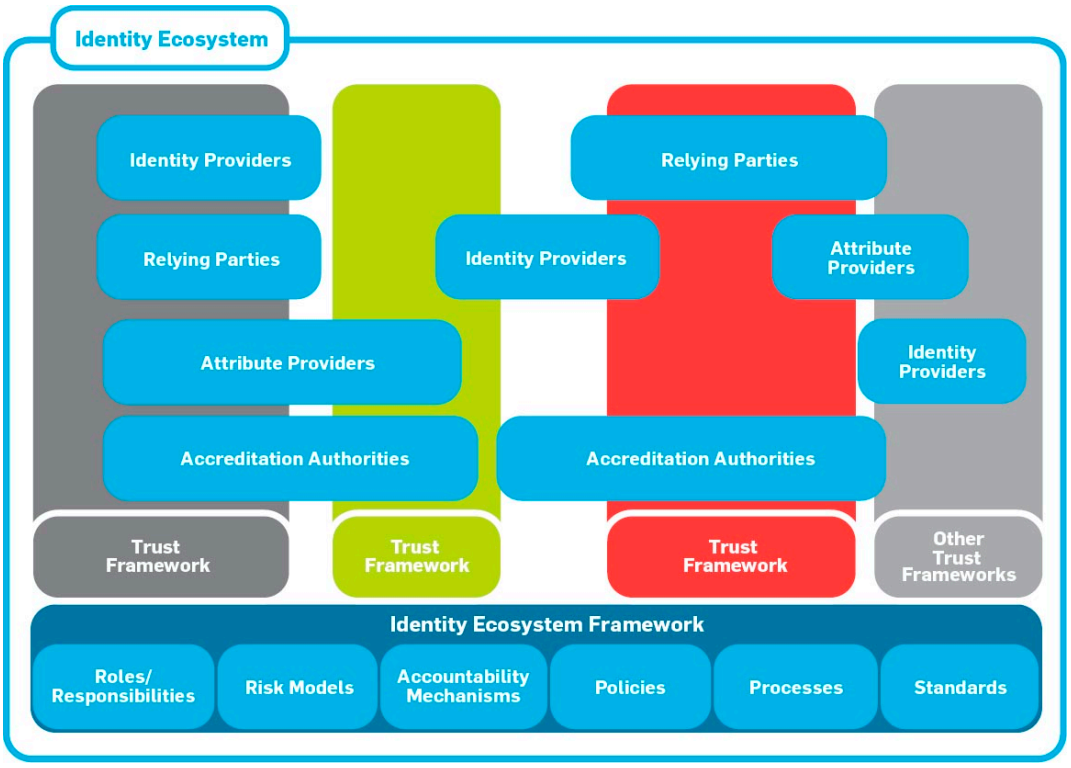

Figure 1.1: Identity Ecosystem

Identity Ecosystem Framework (Figure 1.1) is the structured set of rules, policies, standards, models, requirements and mechanisms for creating entire environment for participants. The steering group plays the administration role, which will ensure the accreditation authorities for the participating parties, bodies, and organizations to be the part of ecosystem. The third component is the Trust Frameworks, which are developed by community, organizations, and individual with similar goals of information exchange. These components already have some certain standards, policies, process and procedures that provide certain level of trust in online transaction. It could be used as baseline or can be incorporated inside the identity ecosystems to develop another level of trusted frameworks policies. The fourth component assures the use of Trusted Frameworks and validates their trust processes and mechanisms. The last component is to determine the service provider passivity with the Identity Ecosystem Framework [36].

1.6 Trust Framework

A Trust Framework is a flexible concept developed by community members with similar goals and perspectives [36]. It outlines the rights and responsibilities for all the participants in the Identity Ecosystem. It states the policies and standards processes and procedures specific to the community. To be specific, Trust Framework is the set of requirements for the applicants to be certified, where Trust Framework Providers manage certification and Trust Framework Operator providers manage to connect required certified participants and services. Different trust frameworks can exist within the Identity Ecosystem, and sets of participants can tailor trust frameworks to meet their particular needs. In order to be a part of the Identity Ecosystem, all Trust Frameworks must still meet the baseline standards established by the Identity Ecosystem Framework [36].

This research has to find or derive a reference model that can be referred to develop Trust Frameworks need in Identity Ecosystem. Thus, agility, interoperability, system and services will be the most important perspectives for evaluating each enterprise architecture frameworks. The research analysis is performed to review the current

status of the major enterprise architecture frameworks enterprise architecture methods, processes, domains, roles, artefacts, and tools perspectives.

1.7 Enterprise Architecture Framework

The existing industry has many frameworks available for referencing their strategies, policies, processes, models, domains and tools although they may have different purpose, goals and domains. They can be used as referencing model, if they provide every aspects of the framework modelling to build Trust Framework. This Trust Framework can be referenced from different existing concepts of framework building models, if they can satisfy the requirements of Trust Framework and its Development process. In the search of reference model, enterprise architecture frameworks could be the orientation baseline for framework development. Enterprise architecture frameworks are the guidelines to design the future strategies of business ideas and policies. They can be analysed, compared, measured and validated to build the Trust Framework required in Identity Ecosystem. The contemporary industry has several enterprise architecture frameworks that can provide the similar kind of solution, architecture and components.

According to ANSI/IEEE STD 1471-2000, architecture is defined as the "fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution" [8]. Enterprise Architecture is an organizing logic of analysis, design, planning, and implementation of business process and IT infrastructure to integrate and standardize the requirements of organization operating model. The well-organised system includes the architecture principles and information design, process and business logic to achieve successful organizational benefits [2]. MIT CISR defines enterprise architecture as "the organizing logic for business process and IT capabilities reflecting the integration and standardization requirements of the firm's operating model" [3]. This explains the enterprise architecture is a strategic guide to define the vision of the firm operation to align IT and business ideas for deriving business value.

Enterprise Architecture is a guide to define the principles, practices, process, strategies, tools, models, and taxonomy to create architecture description of the system. The principal guidelines divide the architecture descriptions in different domains, layers, and views. Diagrams and metrics are also documented and defined in enterprise architecture framework. There are several matured enterprise architecture frameworks, which have clear definition of business architectures, data architecture, technology architecture and solution architecture. This will help to design the new frameworks or improve the existing frameworks with updated policies and strategies.

After the comparison of enterprise architecture frameworks, the research question for the development process comes in front. Most of the enterprise systems are similar in nature and provide features that are required to build the reference model. The issue here is the process for developing the framework. After the selection of enterprise framework, there is a need of model or method to develop the enterprise system with adaptive feature that has ability to serve the customers with value. So finally, we need another model to develop the framework with agility.

1.8 Adaptive enterprise Service System Model (AESS)

Gill Framework developed by Dr. Asif Gill comes in as a solution for developing agile and interoperable enterprise systems. Gill Framework provides the adaptive enterprise requirements management capability reference model to manage integrated adaptive enterprise strategies, architectures, projects and services. This model can be tailored with any specific enterprise architecture framework with inheritance of agile and non-agile requirements, management practices, tools and techniques. This framework includes The Adaptive enterprise Service System Model (AESS) to define the enterprise context on agility, system and service science, which is called theory Triangulation [31]. The AESS conceptual model can be used for describing the overall context and scope of an adaptive enterprise. Enterprise architects need to use the enterprise context and scope as a guide for developing the adaptive EA capability

and artefacts. The AESS conceptual model adopts the "Theory Triangulation" approach [60]. The AESS model describes and analyzes the adaptive enterprise context and scope through the lens of three well-known theories: agility [61], living systems [62], and service science [63]. With the combined concept of agility, living systems and service science, Dr. Gill have tried to implement the AESS model to the adaptive system development model for enterprise. This concept could be the final touch to this research to design the reference model for the Trust Frameworks.

Thus, AESS model of Gill Framework can be used with the selected enterprise Architecture Framework to develop adaptive enterprise system. This combined process model can be defined as a reference model for developing Trust Frameworks in Identity Ecosystem.

1.9 Overview of the study

The following chapters provide details of the history, background, analysis, reference model and results.

Chapter 1 is the introduction to the research, which covers the context and objective of the study. This explains the background and describes their origins and uses. It gives a brief idea of Identify Ecosystem, Trust Frameworks, the architecture frameworks and adaptive development process.

Chapter 2 is the literature review about previous works done in the similar domain. This explains works already done in Identity Ecosystem and the comparison of existing enterprise architecture frameworks.

Chapter 3 is about the research methodology used to verify the findings and outputs. It speaks about the methods followed in this research.

Chapter 4 explains the requirements of Trust Framework and its attributes.

Chapter 5 is the measurement process for all attributes and aspects of enterprise architecture frameworks.

Chapter 6 is the analysis basis for comparing enterprise architecture frameworks. It mentions the criteria for all the attributes.

Chapter 7 is all about the enterprise architecture frameworks. It lists all the top most frameworks from the industry and compare list. This comparison analysis consist fourteen enterprise architecture frameworks, which is never done before in same research domain.

Chapter 8 is the comparison assessment and evaluation to explain each attribute in detail.

Chapter 9 suggests the final solution for the reference model including Gill framework to develop adaptive enterprise system and shows the result with its discussion for the conclusion.

Chapter 10 is the final conclusion of the research and future perspective followed by references.

Chapter 2 Literature review

2.1 Introduction

This section is the state-of-art of original concept of Identity Ecosystem, Trust Frameworks and enterprise architecture frameworks. There were limited resources on the concept of Identity Ecosystem itself and its Identity Framework. An identity ecosystem [36] was proposed to mitigate identity theft, fraud, and digital crime from its root. As the existing systems are based on different identities to store the user's information, its difficult to identify the real users and their personal information linked in different systems. The proposed system itself is brilliant idea but it's not easy to manage each digital identity for everyone in this world. The proposal defines several relevant terms with identity, mission, strategy, governance, policy, services and service systems.

However, NSTIC [36] succeeds to be a strategy to review the existing service system and propose a new identity management system. The service provider is the Federal Government and the stakeholders are the service clients. And the key strategy is to define methods to increase the level of trust bond coupled with digital identities between service provider, server system and service users. All the individuals, organizations, services, and digital components are connected and interact through a trusted cyber ecosystem to enhance online choice, efficiency, security, increase privacy, ease of use, confidence and innovations. The overall objective is to increase protection of individual privacy in digital world [36].

The review [46] gives brief idea of both version of original proposals and its cyberspace policies. The review concluded the idea to be a well-conceived vision for future operations in global cyberspace and it can solve the security problems created due to limited infrastructures, buggy code and careless users. But the end of the review outlines the issues to verify the Trusted Identity Authority,

which provides the Identity.

The solution could be the reference model or framework must have core ability of adaptive enterprise for service system. An agile service-centric system of service systems is the opposite of traditional collection of product centric, which are value-propositional rather than value deliverable and value co-creative not only in business units [47]. Dr. Gill [47] refers enterprise agility as an ability to handle changes in enterprise by management and can be defined in terms of five attributes of agility.

- Responsiveness
- Flexibility
- Speed
- Leanness
- Learning

And these five principles can be used to measure agility level of an enterprise. Although the measurement can be done in any units, it has to reflect the presence and absence of agility feature. To measure the adaptive nature of an EA framework, comparison of the major enterprise architecture frameworks are essential.

The literature study started with the focus on enterprise frameworks and its comparison from different aspects taking base of various attributes of enterprise architecture frameworks. This literature study began for the purpose of reviewing major enterprise architecture frameworks from methods/processes, domains, roles, artefacts, and tools perspectives concentrating on security architecture level. The study included the most used enterprise architecture frameworks to the evolving frameworks with their comparative analysis of each architecture development methodologies. The research required reading their development models, process, methodologies and principles from origin to the current state of the maturity. This review includes their documentations, books, papers, journals and Internet articles based in the research of enterprise architecture frameworks.

The study addresses the research questions based on the state-of-art of enterprise

architecture frameworks that allows highest level of data security and interoperability in cyberspace.

2.2 Previous related works

All the resources for literature review were taken from the IEEE papers, authorized research institutions, developer's official published materials. Enterprise architecture frameworks that were studied, researched, reviewed and already used in industry were included in the comparison list. Their maturity level and depth of the each framework development level to be adaptive enough for the enterprise changes were described to get a complete view of enterprise architecture. Comparisons were done before for different purposes. There are the papers and research works done to compare their attributes to verify the strength of the listed frameworks. Each framework is build for its own purpose regarding its usage. The government frameworks are more specific to domain whereas the commercial tries to cover the most relevant areas.

The comparison [9] compares five top enterprise architecture frameworks with introduction model of understanding for AF based on fundamental elements of architecture. AF Model could be selected or tailored for system or enterprise architecture development in specific environments or multiple frameworks could be used in conjunction to meet particular development needs. Susanne Leist and GregorZellner [34] evaluated the current enterprise architecture frameworks to find the contribution to support architecture development projects [34]. It uses Specification Document, Meta model, Role, Technique and Procedure Model for the base of analysis [34].

The comparison was done to choose the best EA framework in the paper [47] but it concluded with the result of "NO" best framework and none of them are fully complete. Each of them has own strengths and weaknesses. They are adaptable but selection should not consume huge amount of resources [47].

Comparison of enterprise architecture frameworks based on EA quality attributes is done in the paper [48] to select suitable framework that meets the quality requirement. This analysis compared the linkage of architecture model and system development life cycle, and also found that there is no such single EAF that can support all quality attributes.

Furthermore an evaluation is done to find if any enterprise framework is suitable for E-Governance but the result is similar as usual [49]. No Single framework can be selected as a solution for all E-Government requirements [49]. It has covered issues like semantic interoperability, service and design principles [49].

Urbaczewski [50] compared four frameworks from views/perspective, abstraction and development life cycle to find best-fit framework to the stakeholder's needs for the project. Although it mentions the need of the specific framework for each domain it doesn't mention detail content for choice of framework.

The impressive work in comparing enterprise architecture frameworks done by Mahesh and Shantanu [59] has detail measurements of each attribute both functional and non-functional. It has setup its own measurement unit to measure each of these attributes but the result just compares the attributes rather than selecting among them. The final note mentions that EA framework selected depends on organizational culture, mission and initial principles to architecture principles in implementation phase [59].

This similar comparison analysis was followed by another research paper [51] to purpose a scheme for selecting EA Framework systematically. They purposed a model for it, bit of calculations and final formula for the selection. The comparison was done in detail including every aspects of the enterprise architecture framework. Although it mentioned that the followed approach is better than any of the past methods, it not mentioned clear verification of the results and if it can be used to for any features of adaptability. But both of these measure them in so much depth, that it's even harder for any organisation to

choose the best-fit architecture. The comparison tool should make the selection process easier not harder to decide which one to follow. This led to confusion of the measurement. The measurement units should narrow down to “YES” or “NO”, if any selection method wants users to use the selection tool for selecting EA framework. If the purpose is just to measure the attributes and show their strength and weakness for specific nature of the EA framework, then the measurement method comes to be most effective.

The most used common resources for comparing enterprise architecture frameworks by all reviewers is comparison done by Roger Sessions [52]. He compares four top enterprise architecture frameworks with their complete description with strength and weakness. Good news is that these frameworks can complement each other but the bad news is the difficulty to choose one over others. As they have specific natures, they are hard to compare from single perspective and have so little in common.

From the above reviews, each of these reviews was done for the different specific purpose of the requirements. They were done for the research and analysis of the finding the strength of each framework for required domain. The reviews were based on their own attributes and domains, although the similar major frameworks were included in comparison. The major conclusion from the previous works was the limited number of frameworks in each review to conclude the selection for reference model. Though the reviews were not specific this research title, they comparison part had maximum number of 5 to 7 frameworks. All of them lack the major framework like Oracle enterprise architecture framework, which has much more potential to be the reference model for the Trust Framework.

Chapter 3 Research Method

The objective of this research is to suggest a reference model to develop Trust Framework for Identity Ecosystem. It requires the comparison of enterprise architecture frameworks to select most relative method to under go AESS model of Gills Framework for the agility development process.

The review was started with the original concept of Identity Ecosystem to the reviews of most used enterprise Frameworks in industry, which are matured and followed by many enterprise architects. The research method was started with problem in building Trust Framework to its solution. It can be summarized as below:

Step 1 - Identifying the research problem in original concept of "*National Strategy for Trusted Identities in Cyberspace"*. The research problems and objectives for this research have been identified based on the literature review of concepts and enterprise architecture frameworks, which were mentioned in Chapter 1.

Step 2 - Extracting comparative concepts: This research applied reviewing and analysing the most well-known enterprise architecture frameworks, best fit for referencing model of Trust Framework. The major argument for comparing the enterprise methods/processes, domains, roles, artefacts, and tools offered by different frameworks has been identified in chapter 3.

Step 3 - Integrating the concept of agility, system, service science and interoperability with the most relevant enterprise Architecture Framework, selected from the comparison. The solution for developing adaptive enterprise system was to integrate AESS conceptual model to develop Trust Framework with enterprise architecture.

This literature study in Chapter 2 outlined the problems for developing Trust Frameworks used for the interaction between Trust providers and participants. The required solution should have proper guidelines, policies and roadmap with agility capability. This required the review of enterprise architecture frameworks from methods/processes, domains, roles, artefacts, and tools perspectives. The study included previous works on comparative analysis of enterprise architecture frameworks with methodologies followed. This review included previous available papers, documentations, books and Internet articles. The study addressed the research questions based on the state-of-art of original concept, Identity Ecosystem, Trust Frameworks, enterprise Architecture Framework that has highest level of agility and interoperability.

3.1 Research Questions

- *Which existing enterprise architecture framework could be the possible reference model based on the attributes?*
- *What are agility, interoperability, service science and systems thinking concepts for enterprise architecture frameworks?*
- *What could be the possible solution to develop adaptive service systems from agile enterprise architecture frameworks?*
- *What could be the possible solution to develop Trust Frameworks in Identity Ecosystem proposed by "National Strategy for Trusted Identities in Cyberspace"?*

With respect to the first research question, fourteen enterprise architecture frameworks were compared on their enterprise architecture attributes with their aspects/perspectives to analyse their depth of enterprise architecture planning.

The second question addresses the requirements of the research. The relevant enterprise Architecture Framework must be able to provide agility, interoperability,

service science and systems thinking concepts to meet the requirements of the Trust Framework.

The third question provides the solution for developing adaptive enterprise system from selected enterprise architecture frameworks. This offers the model of developing adapting reference model for Trust Framework.

Finally the last question is to get the result and conclusion on the research topic. The final result is suggestion for the reference model that can be used to develop different needed Trust Frameworks in identity ecosystem purposed by national strategy for the trusted identities in cyberspace.

3.2 Search process

The search of enterprise architecture frameworks started with the collection of developer's documentations, review papers, case studies and white papers. This included vendors' or developer's research papers and reports. The search continued to conference papers, journal, reviews and articles published by research centres and independent researchers. This technique is followed before by the most of the comparing articles and papers [25][48][50][51][59].

Research papers and documentation were selected for each framework that include either empirical studies or detail description of their architecture levels. The search process has list of few research articles for the comparative analysis of the enterprise architecture frameworks, which is extended to more number of frameworks.

	Titles of Papers	**Publisher**
1.	National Strategy for Trusted Identities in cyberspace	Whitehouse
2.	Essential layers, artefacts, and dependencies of enterprise architecture	IEEE
3.	A comparative analysis of architecture frameworks	IEEE

4.	Interoperable enterprise systems: principles, concepts, and methods	ELSEVIER
5.	A review of enterprise agility concepts, frameworks, and attributes	ScienceDirect
6.	Enterprise information security, a review of architectures and frameworks from interoperability perspective	ScienceDirect
7.	Enterprise Security Architecture	SABSA
8.	Applying Agility and Living Service Systems Thinking to enterprise Architecture. International Journal of Intelligent Information Technologies	IJIIT
9.	Enterprise information security architecture a review of frameworks, methodology, and case studies	IEEE
10.	A Framework for Evaluation of enterprise Architecture Implementation Methodologies	WASET
11.	A Comparison of the Top Four enterprise-Architecture Methodologies	Microsoft
12.	Analysing the Current Trends in enterprise architecture frameworks	AEA
13.	Challenges to Building a Global Identity Ecosystem	The Open Group
14.	Managing information security in a business network of machinery maintenance services business – enterprise architecture as a coordination tool	ScienceDirect
15.	Comprehensive measurement framework for enterprise architectures	IJCSIT
16.	A Comparative Analysis of enterprise architecture frameworks based on EA Quality Attributes	IEEE
17.	Applying Agility and Living Service Systems Thinking to enterprise Architecture	IEEE
18.	A Holistic Approach for enterprise Agility	Information Science Reference
19.	An enterprise Security Program And Architecture To Support Business Drivers.	Technology Innovation Management

		Review
20.	An Evaluation of enterprise architecture frameworks for E-Government	IEEE
21.	A Scheme for Systematically Selecting an enterprise Architecture Framework	IEEE
22.	Evaluation of Current Architecture Frameworks	ACM
23.	Towards a Framework for enterprise architecture frameworks comparison and selection	Google Scholar
24.	Ontology for Characterising Architecture Frameworks	Springer
25.	A Comparison Of enterprise architecture frameworks	Google Scholar
26.	An overview of enterprise Architecture Framework Deliverables	Google Scholar

Table 1: Selected lists of research papers

3.3 Inclusion and exclusion criteria

The research resources were based on different factors for filtration. Resources based on the above frameworks were included, which are the core documentations of the frameworks explaining their development architecture, layers and views. The research papers that are related to agility, service science, interoperability, principles, concepts, and methods are included for this research. The comparison methods, measurement processes and units were included for detail proof. Papers with more number of perspectives were used for the comparisons rather than those, which are just theoretical description [25][48][50][51][59].. The works that have relevant methods and objectives were included. The frameworks that are most reviewed, recommended by industry and developed by enterprise venders are included.

3.4 Data extraction

The listed attributes were extracted from each articles and resources used in research.

- Type of reference (journal, conference, workshop)

- Research objective
- Evaluation methodology for comparison (experiment, case study, etc.)
- Evaluation metrics and definitions for comparison of enterprise architecture frameworks
- Evaluation results and limitations

3.5 Quality assessment

All the resources for research were taken from authorize official research institutions, IEEE papers, developer's published materials, documentations, whitepapers and guides. The frameworks that were studied, researched, reviewed and already used in industry were in the list for the comparison. Their maturity level and depth of each framework development level were described enough to get a complete view of the enterprise architecture.

To conclude this section, research process was much dependent in literature reviews, enterprise architecture frameworks and search of model to develop agile enterprise systems. This process required reviewing of many frameworks and its development process.

Chapter 4 Architectural Attributes for Trust Frameworks

4.1 Introduction

With an ability to provide the services, being an adaptive enterprise system is the key parameters while designing Trust Framework for Identity ecosystem. It should be public user centric, seamless, responsive and secure system that can allow the users to interact with any enterprise applications without going through the complex structures of functioning. The user and system interactions should be done in secured process that keeps the whole process as simple as possible. This convergence of the system requirements needs a major re-organization of current processes transform and can organize the changes in future. Even though the different organizations and departments have different process, procedures, work practices, rules and regulations, they have to follow the common fundamentals along with shared infrastructure and values via enterprise Architecture Framework.

While designing the Trust Frameworks in Identity ecosystem, it is very important to identify the design requirements that play an important role in the selection of enterprise Architecture Framework or design new reference model to build Trust Frameworks. Again, the Reference Model has to be adaptive enterprise Architecture Framework that stands over the concept of service science and system thinking. After the description of basic elements for selection of enterprise architecture frameworks, the development process of Reference Model needs to support agility, service science, system thinking and interoperability.

The concept of agility, service science, system thinking and interoperability together transforms an enterprise to adaptive enterprise. If these adaptive features could be found in any of the above listed enterprise architecture frameworks, it can be strong attribute for the selection of reference model for Trust Framework. The agile EA has

to be agile human-centric, service-centric, value propositional and value co-creative of living multi-agent services systems, which has the ability to scan and sense the changes, form an appropriate quick and flexile response, and learn from their knowledge and experience [31].

4.2 Agility

The concept of agility is the process of adapting the changes of states. The process of coping with uncertainty situation, requirements and changes of system is agility [6]. As the businesses are flexible in nature, Reed and Blunsdon (1998) described organizational flexibility as an organization's capacity to adjust it's internal structures and processes in response to changes in the environment. And the same concept of agility in enterprise is defined by Kidd (1994) as a rapid and proactive adaptation of enterprise elements to unexpected and unpredicted changes [21]. According to Haeckel, the behaviour of flexibility can be a four-phase adaptive phases; first, sense situational or environmental transformation followed by interpretation of changes for unexpected threats, third is how to respond and finally fourth is all about an act to make the flow in normal state [43]. This concept requires the effective management of knowledge, faster learning ability to extract data and understand the analysis. The next process is successful decision making for robust deployment of proper solution [43]. Enterprise architecture frameworks that have agility features will be selected as the matured, as it can provide solution for any uncertain ability. Dr. Gill [31] mentions agility concept as "adaptive extended enterprises". As Modern enterprises have to do take themselves beyond their certain boundaries and requirements for the existence in ecosystem, which includes customers, partners, collaborators and community, they have to be adaptive in nature. If any enterprise architecture frameworks support agility, it will be suitable for the development of reference model because it can adapt the changes in its environment. In analysis summary, Dr. Gill define agility as:

''An entity is said to be an agile enterprise when an enterprise is responsive (scans, senses and reacts appropriately to expected and unexpected changes), flexible

(adapts to expected or unexpected change at any time), speedy (accommodates expected or unexpected changes rapidly), lean (focuses on reducing waste and cost without compromising on quality), and learning (focuses on enterprise fitness, improvement and innovation).”

Furthermore agile enterprise is the ability to handle the expected and unexpected changes that can be expressed in terms of five concepts of agility: responsiveness, flexibility, speed, leanness and learning [31]. Dr. Gill uses these five agility principles to measure the degree of agility of an enterprise. This principle can be used to differentiate a non-agile enterprise from an agile enterprise. These are important attributes to enterprise, which are described further in following section.

Responsiveness: This is the most important capability of the adaptation in enterprise system. It is ability to scan and sense the unexpected or expected changes in surrounding situation to response opportunities internally or externally. Responsive enterprise always takes action in different situations rather than remaining silent.

Flexibility: This capability lets enterprise to adjust to expect and unexpected changes. It provides enterprise to allow space for change in business requirements and demands. The reference model should be flexible enough to cover the business requirements in development stages and future.

Speed: This feature is a quick behaviour of an enterprise to the situation occurred. The ability to deliver fast solution for the situational changes in business process defines the speed of an enterprise system. The reference model should provide fast response to the changes occurred in different standard business processes.

Leanness: This is ability to maintain the quality of result with minimum available resources. It refers to compactness and tidiness of enterprise to keep the quality of the output same with marginal usage of assets. The reference model has to lean the resources to give maximum expected output.

Learning: This power makes enterprise to keep itself up-to-date. The learning is the continuous growth and adaptation of knowledge, skills and past experiences over the period of time. The reference model should be able to communicate the past experiences for similar problems in future.

Agility in an enterprise is the ability, but to extend its understanding the concept of system thinking and service science required.

4.3 System thinking concept

The process of problem solving, by viewing problem as overall system that adds to further development of unintended consequents is the system thinking concepts. Bray Richmond in 1994 defines Systems Thinking as the art and science of making reliable inferences about behaviour by developing an increasingly deep understanding of underlying structure and further explains Systems thinking is not one thing but a set of habits or practices [35]. Later in 2007, it was defined in more detail structure by researchers as an idea that can influence the existing concepts, theories and knowledge. Systems thinking as an idea permeates both popular culture and numerous scientific fields including: planning and evaluation, education, business and management, public health, sociology and psychology, cognitive science, human development, agriculture, sustainability, environmental sciences, ecology and biology, earth sciences, and other physical sciences [20]. It is different than then system science but its something one gets as a consequence of relating simple rules based on patterns of thinking. And if this concept of system thinking can be used in enterprise architectures, the system can solve the problems from the previous repeated processes. Dr. Gill researched this concept to build the adaptive enterprise with enterprise architecture framework. He implements the system thinking concept of living multi-agent system or living organism (e.g. like human), which is autonomous, interdependent, integrated, context aware, adaptive, self-organizing, and has lifecycle in an agile enterprise [31]. This is the process to learn from past experiences and tries to implement them to solve further more problems. The adaptive enterprise architecture should have following eight points System,

Autonomous, Interdependent, Integrated, Context Aware, Adaptive, Self-organizing, and Lifecycle, which are the elements of system thinking [31]. Although there are traditional tools as strategic decision making in existing EA, the agility or independent decision-making could be another biggest achievement that can help system to be more agile and adaptive. According to Dr. Gill in his research [31], mentions eight underlying concepts of an enterprise as a living multi-agent system or living organism which are discussed below:

System: An enterprise system is the one that accepts inputs from different sources and processes these inputs under the standard constraints to produce set outputs. System processing performance depends on factors like culture, standards and environment changes.

Autonomous: The concept of autonomous signifies the low-level agent systems that processes inputs and responds to situation in their own way. It is an autonomous aspect of an enterprise system to make process faster and leanness.

Interdependent: This refers to basic units or cell functions of an enterprise system that are integrated and aligned to survive and succeed on expected business goals.

Integrated: The open, non-linear and holistic enterprise system is the complete form of integrated multi-agent system. Many business units, functions, and cells are architected, managed, connected and finally collected to form a single piece of enterprise.

Context Aware: This refers to ability to scan and sense continual changes of the surrounding environment of enterprise business process.

Adaptive: This is the power to strive and maintain the enterprise's internal steady state even though the external business attributes are changed.

Self-organizing: If any enterprise has the ability to adapt continuous evolution and changes toward higher levels, then it's self-organizing. The process involves the communication and feedback to organize its changes.

Lifecycle: This is the ability of an enterprise system to create, use, maintain, transform and expire the business process over a certain time period.

Designing complex adaptive enterprise involves a system with ability to be autonomous, integrated, context aware, adaptive, self-organizing and should follow complete product life cycle for any process. The reference model for the Trust Framework should have above attributes to design the required Trust Frameworks. The adaptive enterprise is a system of service systems with better understanding for context and scope of real services, which is discussed as the "Service Science" concepts in following section.

4.4 Service science

The process of creating values for consumer and producer with a specialized knowledge or skills is defined as service [44]. In the current world, people are connected with social and economical factors. And the connection is being stronger with the evolution of advanced and new technologies that offers services to every customer through global value chain [44]. However, the services are developed and delivered to the customers that may or not meet the customer requirements 100%. But still the service operates its network, which may not be compatible within the time change, as the requirements of the customers are very dynamic in nature with time, social and economic factors [44].

The concept of social-technical system [44] developed to create values for both provider and consumer is a service system. This service could be business, idea, innovation or just a product that is built for every user around the globe and it meets the requirements of majority population. The service systems should be user-centric, informative and focused in user satisfaction. The paper [44] cites "Indeed, almost

anything from people, object, to process, for any organization, large or small – can become digitally aware and networked." by Palmisano.

In similar concept, an enterprise has to build their structure with users as a centre of their service. The structure has to cover the social and cultural values with collaborative digital innovations of services to satisfy its customers around the world. As whole, service science builds service with new approach with their complementary components for future. It's not easy to improve the quality of life and sustainability of existing and future service systems. But the coalition of traditional disciplinary, sectorial research and practice around the service system can be a beginning step [45].

This concept of service system can be used in enterprise architectures. Dr. Gill [31] mentions that this concept can be used for an adaptive enterprise system as a combined echo-system of "service systems". And also defines the modern service science theory as mutual volunteer interactions between organizations or service providers and consumers offering mutual benefits [31]. This concept of service science can be used to define the adaptive nature of service system with its conceptual elements. Dr. Gill mentions eight conceptual elements as Offering, Use, Abstraction, Interaction, Access, Governance, Stakeholder and Concern to define the service science in agile enterprise which are described as below [31]:

Offering: This concept is about the collection or inventory of services provided by an enterprise system. Enterprise services offer certain process to solve the problem of users based on it business requirements. The reference model should be able to offer certain services for the customers that create value.

Use: This is a mutual understanding to share a service through different mechanisms like proposal, contract, access and feedback. The reference model should be usable among the different domains or parties.

Abstraction: This represents different types of entities that create the value, mutual benefits and solution from the available resources.

Interaction: This is the value-proposition interaction between service systems to give a certain valuable output and vice-versa.

Access: This is ability to access rights based on different attributes defined by systems.

Governance: This mentions the governance mechanism to take responsibilities in service conflicts or disputes.

Stakeholder: They are the part of ecosystem as well, that plays role to run the whole system. They are service creator, provider, consumer, carrier, partner, auditor, observer, competitor, regulator and broker etc.

Concern: This regards to performance, security, quality and compliance ability of the service system to stakeholders.

Dr. Gill explains the agile enterprise system is based on three major abilities: agile, system thinking and service science, but to design different kinds of Trust Frameworks, there is the need of integration and interoperability. Integration is the collection of units as explained above in section 5.3 under the system thinking concepts. But interoperability is different ability in systems to share processes to enhance the output of another process in the system.

4.5 Interoperability

In the modern enterprise architecture, the enterprise integration is a major issue and especially in e-Governance. The reference model for the Trust Frameworks for identity ecosystems must support the concept of Interoperability to make system secure and adaptive. And it has been a competitive advantage for every system that

supports the interoperability [23]. As mentioned in [55] the Secretary of Defence in April 1998 "joint operations have been hindered by the inability of forces to share critical information at the rate and at the locations demanded by modern warfare." This problem was handled by interoperability based on technical standards such as the Joint Technical Architecture and the use of a defence-wide common infrastructure. They promoted interoperability concept through the framework called C4I (Command, control, communications, computers, and intelligence), which are used by U.S. defence during joint operations. U.S. defence defined LISI (Levels of Information Systems Interoperability) as an ability of systems, units, or forces to provide services and accept services from other systems [56]. In today's world "Interoperability" play a vital role in connecting various business processes among different organisations

As mentioned by F.B. Vernadat [22], interoperable enterprise systems allows a mixed service and process orientation, to support synchronous and/or asynchronous operations, both at business level (business events, business services, business processes) and at application level (workflow, IT and Web services, application programs). This holist approach of integration is the capability of performing interoperation between two or more different entities that could be bits of software, processes, systems, business units, etc. [22]. The reason behind the enterprise interoperability is communication, cooperation and coordination of organizational units or business processes either within a large (distributed) enterprise or within an enterprise network. This technology facilitates to build the integrated enterprise system.

A single enterprise has various administration, business processes, data and systems integrated to provide seamless service offered by that organization. This creates the necessity of coherent information system architecture. But for the distributed enterprise integrated service is much more complicated, as it has to face global scenario of business process without corporate boundaries. There are nodes, which has to be removed from the process, and some of them have to be added to implement new business strategies [22]. Therefore, large and distributed enterprises are more

complex in terms of systems integration, which needs the agility features to be adapted into the environment changes. And Agility requires interoperable enterprise systems, i.e., reconfigurable systems made of systems that can work together [56].

Although Interoperability has been implemented in wide industrial sectors, but based on [22] and in reference to the European interoperability framework [57], interoperability occurs in three levels:

- Technical level: for data and message exchange
- Semantic level: for information and service sharing
- Organizational level: for business unit, process and people interactions across organization borders

But in the second version of EIF, European commission in 2010 has mentioned anther layer called legal interoperability to define the cross border information exchange. And [56] have collected the ETSI's approach [58] for interoperability with four levels:

- Technical: secure data transfer technically
- Syntactic: processing received data
- Semantic: data processed to information
- Organizational: processes to link the different systems automatically

The main objective of interoperability is enabling the information exchange between different systems, which can be used for further process. The process needs mutual understanding to achieve this objective. It also needs specific standard protocol to exchange every bit and bytes, descriptions methodology to process data into information. The sharing of information may vary per model used to provide and get information. It also carries information about domains and its agreement. The information exchange process can be standard or can be decided in the user-side but the final process for the result should be as per standard described in the agreement. There are frameworks, models, maturity levels and standards for the interoperability.

This research report will focus on the side of enterprise interoperability and its scope, usability, models, frameworks and standards in different domains.

Interoperability places customers always in the centre of architecture, creates guidelines and follows the standards. It is a flexible operation to transfer information in the heterogeneous systems to provide mutual services. The frameworks, models and standards are still under development; they are used as base for each other and move towards the matured models. Designing adaptive enterprise system needs the interoperability as a major feature of reference model to develop Trust Frameworks.

Chapter 5 EAFs comparison process

5.1 Introduction

After literature reviews and previous works, research requirement was to compare different enterprise architecture frameworks. The process for selection was started first, which listed out all the possible frameworks on base of different aspects of comparison requirements mentioned in 6.2. Then section 6.3 explains the measurement process and finally the attributes to be measured are described in 6.4.

5.2 Selecting enterprise architecture frameworks

The selection was done in the base of attributes required for Reference Model to design Trust Frameworks. The selection was done in longer list than ever compared before. Usually, previous works have listed four to six number of major enterprise architecture frameworks, but here the list has increased to fourteen so that the possible enterprise architecture frameworks could be matched among them. The list includes all the major frameworks used before in different industries, domains, private and government sectors to achieve their own goals. Each of them has their own goals, objectives and specific architecture based on; where they are intended to be implemented. This list includes the following major enterprise architecture frameworks:

- Zachman Framework (ZF) [10][12][13]
- The Open Group Architecture Framework (TOGAF) [15]
- Department of Defence Architecture Framework (DODAF) [16]
- Federal enterprise Architecture Framework (FEAF) [17][18]
- Gartner's enterprise Architecture Framework (GEAF) [30]
- Treasury enterprise Architecture Framework (TEAF) [17][38]
- Reference Model of Open Distributed Processing (RM-ODP) [33]
- British Ministry of Defence Architecture Framework (MODAF) [32]

- 4+1 Architectural View Model [24]
- Extended enterprise Architecture Framework (E2AF) [29]
- SABSA (Sherwood Applied Business Security Architecture) [19][23]
- The Oracle enterprise Architecture Framework (OEAF) [4]
- Generalised enterprise Reference Architecture and Methodology (GERAM) [41]
- Model Driven Architecture (MDA) [42]

They were selected on the base of their maturity levels, usage, implementation, rating, reviews, documentations and references available for the research. These listed frameworks clearly mention their philosophies, dimensions, structures, artefacts, development process, lifecycle, tools and non-functional requirements perspectives. These attributes are described and defined by various developers and architecture designers in their development documentation. Some of them are reviewed in previous works as well. There is also framework like Oracle enterprise Architecture Framework, which is never reviewed by any reviewers. The vendor and the frameworks have published enough materials to understand it and its architecture analysis looks powerful as well. This should be the first time where the list of thirteen enterprise architecture frameworks compared with their attributes in a single table. After the list was ready, the measure process has to be defined. There are various methods and analysis tools to measure these attributes proposed by various designers and architects.

5.3 Measurement process

Measurement is the process of describing entities in numeric values or symbols. The measurement denotes the distinctive characteristics that are preserved in each identified entities as defined in [59]. This mentions entities, attributes and rules for assigning values to the attributes. These rules can be used to define measurement scales. Although there are various methods and scales for measuring these enterprise architecture frameworks attributes, it may not be normal to scale them in wide scale of numbers. There are also methods using symbols or alphabets and numeric digits.

Both the methods describe same distinctive characters in each framework. The most important denotation for measurement process is to show the features are present/supported or not present/not supported. Whatever the symbols are, the meaning remains standard.

Measurement Entities	**Scale values**		
Present/Supported	2	F	YES
Partially Present/Supported	1	P	-
Not Present/Supported	0	N	NO

Table 2: Measurement scale

Any symbols can be used to scale the entities, but in this case, there are methods to scale the entities value from 0 to 5, which helps to detail the analysis but it makes harder to decide, if they support or not. Even though the measurement analysis are great to describe their detail properties, it would be easier to compare if they support or not, Yes or NO. This approach would be much better to deicide whether the entities are present in framework or not. But to keep the research theories consistent to previous works, here the measurement follow similar scaling of 0, 1, and 2 for NO, PARTIAL and YES respectively as mentioned in Table 2. The measurement process starts with the scaling score values from 0 to 2 as per figure above.

Although it is difficult to find the exact presence of each attributes in each framework, this measurement scale range the similar properties through different perspectives. This is the reason why all the possible attributes need to be measured and viewed from different available viewpoints. Each framework has its own architectural views, methods, tools, artefacts, models, roles, domains and processes which are totally different in most of above listed frameworks until they are derived or referred from one another. Measurements of functional requirements (views, methods, tools, artefacts, roles, models, domain, process, abstraction and goals) are

easier as they are defined and mentioned in their developer's documentation or any authorized research papers. They are mentioned in the particular section of each framework if they have functional requirements or not. All the frameworks were studied and based on the research papers, previous works, reviews and recent published papers, the attributes were described and measured within the above derived measurement scale.

The non-functional requirements are specification of functions or services offered by the system. They are the parameters to represent characteristics of frameworks. This comparison of non-functional attributes is very important to this context of study as it defines the attributes from chapter 5. This comparison concludes the selection of framework and solution for the adaptive development method. Designing adaptive enterprise system is not enough for this research goal; the purpose of the study is to suggest a reference model to develop Trust Frameworks from the selected enterprise Architecture Framework.

To conclude this chapter, measurement list of enterprise architecture frameworks and measurement process were defined with scaling to be used for the attributes to be measured for the comparison. This process supports the whole research to find the proper scaling to measure the attributes followed by the attributes chapter 6.

Chapter 6 Analysis base for enterprise architecture frameworks comparison

6.1 Introduction

The comparison processes includes the list of perspectives/aspects based on listed attributes to be compare for selecting proper enterprise Architecture Framework. Perspectives/Aspects are the comparison criteria that focus to define each specific properties, attributes, principles or structures. These perspectives are the basic guidelines or criteria could be present or not in all of them but they defines the completeness and maturity level of enterprise Frameworks. The previous works chapter have already mention these comparisons but these attributes are based heavily on papers [38], supported by paper [59] with papers [25] [49], which explains the process to measure, compare and discuss the selection process of enterprise architecture frameworks. These attributes are all important to find the maturity levels, architecture analysis, levels and verify the elements required for any enterprise systems. These attributes are essential for the building any kind of enterprise systems, which is here, Trust Frameworks in Identity Ecosystem.

6.2 Attributes for comparison

All these listed attributes in Table 3 are defined and described in details to make the clear on them and their needs for the building reference model for Trust Framework. All the attributes are not compulsory to be present in each framework but their aspects have to be verified. Each attributes may not be same in all frameworks, they might have different usage and mean to create different perspectives or aspects. These attributes are based on the different papers and documentations of the enterprise frameworks [7][25][34][38][59].

Attributes	**Perspectives/Aspects**

Views		• Planner • Owner • Designer • Builder • Subcontractor • User
Domains		• Business Architecture • Data Architecture • Application Architecture • Technology Architecture
Processes		
	Goals	• Architecture Definition and understanding • Architecture development process • Architecture evolution • Architecture analysis • Architecture models • Architecture knowledge base • Architecture verifiability • Design trade-offs • Design rationale • Standardization
	Inputs	• Business drivers • Technology inputs • Business requirements • Information systems • Existing architecture • Non-functional requirements

	Outputs	• Business model support • System model support • Information model support • Computation model support • Software configuration • Software process incorporation • Implementation model • Platform • Non-functional requirement design • Transitional design • Design rationale
Guide		• Meta model • Procedure model • Modelling techniques • Role • Specific document • Taxonomy completeness • Process completeness • Maturity model • Reference model guidance • Practice guidance • Governance guidance • Partitioning guidance • Prescriptive catalogue • Vendor neutrality • Information availability • Time to value • Transformation

Artefacts		• Strategy • Process • Application • Software • Technical
Tools		• Business process modelling tools • Data modelling tools • Code development tools • Network design and performance
System Development Lifecycle		• Planning • Analysis • Design • Implementation • Maintenance
Abstraction		• What • How • Where • Who • When • Why
Non-functional Requirements		• Adaptability • Compatibility • Cohesiveness • Conceptuality • Configurability • Consistency • Coupling

		• Diversity • Dependability • Extensibility • Flexibility • Interoperability • Maintainability • Maturity • Portability • Robustness • Scalability • Security • Usability

Table 3: Attributes and perspectives for comparison

Views: Views in Table 3 are interpretations for stakeholder in enterprise architecture to communicate and understand complete architecture. Finally verifies systems to challenge the concerns. Views are the representations of a whole system from the perspective of related set of concerns. Different aspects present in each enterprise architecture framework verify views.

- Planner – it is the perspectives from the planning viewpoint
- Owners – It is the ownership aspects for the system
- Builders – Its is the view of people building the policies and frameworks guidelines
- Subcontractor – It defines the requirements of the small software components
- User – This viewpoint to see how end users use the system.

Domains: This gives a comprehensive view on all relevant aspects of enterprise architecture. It can be a partial representation of an entire system that adopts every concern of all stakeholders. Domains may be called layers or views in different

frameworks but their perspective is same to show the layer of architecture. Usually, they are found in four layers.

- Business – This domain is to address the business information.
- Data – This layer integrate various database component of the system.
- Application - This domain integrates various application and software components in system.
- Technology – This layer addresses the issues for the system users regrading hardware and networking used in system.

Processes: This is the parent attribute for goals, inputs and outputs that defines the development process of the framework. This can be described with the following sub-attributes.

Goals: This attribute defines specific reasons, purpose, targets, objectives and benefits of the process. This gives the clear idea of progress, target dates and motivation to achieve the goal. It starts with the targets to sets the goals, then after manage, plan, prioritize and keep the tracking of each progress. The final step is to make the targets achievable in real world as well. Its aspects are defined below which are very important to understand the depth of architecture frameworks.

- Architecture Definition and understanding – It is defines the architecture frameworks needs identified by stakeholders.

- Architecture development process – It is set of activities performed in architecture construction.

- Architecture evolution - It maintains traceability and variations in system evolution.

- Architecture analysis - It determines the aspects, view and viewpoints of architecture.

- Architecture models - They guide the development plan through proper specification and standards via analysis and design models of system.

- Architecture knowledge base – it keeps the information about designs and decisions to direct enterprise architecture rationale.

- Architecture verifiability – It verifies the set of characteristics to review the services and functions.

- Design trade-offs – It facilitates the alternative selection of rational design for diverse business and technical needs.

- Design rationale – It states proof for verification and review decisions.

- Standardization – It ensure standard for development and architecture are maintained.

Inputs: Inputs in enterprise architecture are outline to integrate business process and goals to offer process components like strategic planning, organizational design, business process reengineering, and systems delivery.

- Business Drivers – They are the inputs as business goals, guidelines, principles, strategies, plans, policies and priorities for the system.

- Technology Inputs –They are inputs as technology platforms, forthcoming architecture, systems interoperability and developing technology standards for strategic architecture direction.

- Business Requirements – This contains requirements of users, functions, data and business process related.

- Information System Environment – This defines the environmental basics of system like budget, timetable, technical constraints, assets and expertise, organizational structure and business knowledge base.

- Existing Architecture – This describes the current standards and infrastructure in system.

- Non-functional requirements – These are Quality Attributes requirement required in system. They are described in another section below.

Outputs: Outputs design guidelines and patterns to direct activities in most efficient way of preferred future. Further more it acts similar to views and models to describe the existing scenario and to plan the future environment. The outcomes may be different in each enterprise systems designed by same framework.

- Business Model – This describes business related models, needs, process, system functions and policies.

- System Model – This model takes the architecture model through critical trade-offs and design decisions for the future system enhancements.

- Information Model – It contains data model, data transformation and data interface

- Computation Model – It contains functional description, process flow, procedures, and software modules with interactions.

- Software Configuration Model – It describes packaging, storage, configuration, management and sharing of software components.

- Software Processing Model – It describes the structure of software processes, threads and run-time environment.

- Implementation Model – It describes physical structure of system like operating environment, hardware components and networking component. It includes the processes for installation, deployment, configuration and management.

- Platforms – This includes operating systems, hardware and networking components, protocols and standards to operate software.

- Non-functional Requirements Design – This model reflects the design of non-functional requirements.

- Transitional Design – It provides designs and plans for system transition and evolution.

- Design Rationale – This is the documentation of design reasons based on analysis and trade-offs.

Guide: This is a proper guidance process to define, maintain, and implement enterprise architecture by managing its lifecycle to produce complete system value for any organization. This is one of the major aspects to compare enterprise Architecture framework.

- Meta model – It specifies consistency of the various architecture artefacts on different layers and in different views.

- Procedure model: This model deploys a modelling language and a modelling tool in EA communication and EA decision processes.

- Modelling techniques: It includes the techniques for the modelling business processes and system requirements.

- Role - It is model for the development and management of the architecture descriptions.

- Specific document – It is a specification of resulting document
-
- Taxonomy completeness – It measure the classification of artefacts in framework.
- Process completeness – Framework guide to provide the guideline in each development process.

- Maturity model – It is a guide for the methodology to assess the effectiveness and maturity of enterprise.

- Reference model guidance – Measures the usefulness of the methods to develop reference model.

- Practice guidance – It guides to develop organizational culture and enterprise architecture mindset in organization.

- Governance guidance – This creates model to understand governance.
- Partitioning guidance – This guides an effective autonomous partitioning of enterprise.

- Prescriptive catalogue – This is a guide for setting up architectural assets and capabilities.

- Vendor neutrality – It checks if framework is neutral to vendors or not.

- Information availability – It defines the availability and quality of information for the framework.

- Time to value – How long it takes to understand and use the framework?

- Transformation – It is an architectural phases like current situation, shot-term or long-term covered by frameworks.

Artefacts: Artefacts are physical documented enterprise architecture components that is produced and shared by system. Artefacts can have associated properties and are represented in a Deployment diagram. Artefacts could be for anything that exists in system. Although, there are not clear categories of artefacts, but to measure the framework capabilities, five of them are mentioned here.

- Strategy artefacts - They document the plans and blueprints of organizational mission.

- Process artefacts – they are evidence of all the processes occurred in system and business cases.

- Application artefacts – they are the diagram for systems interface, communication, systems evolution and web application.

- Software artefacts – they are the documentation for software components used in system.

- Technical artefacts - they are the diagram for network requirements like connectivity, inventory, equipment, building plan etc.

Tools: These are the software tools that help to design the enterprise systems. These tools offer management for all enterprise architecture processes and methods. They help to build system faster and effective. They are usually supporting software tools used in system development process. There are a lot of tools used in every phase of development, but few of them are listed to measure the tools offered by frameworks.

- Business process modelling tools – These tools help in business process modelling and not all frameworks offer such tools.

- Data modelling tools – Tools use to populate the data in the system in development phases.

- Code development tools – Tools offered by frameworks to help in writing code in faster, secured and follow the software development patterns.

- Network design and performance – tools used for designing the network and optimal plan.

System development life cycle: This process is to standardized design methodology with defined artefacts, processes, roles and responsibilities. This includes the development process to deign fasters, clear handoffs and reduce cost. System development life cycle process could have different stages but usually, they should have following development phases.

- Planning – this phase is differentiated for the strategy planning.
- Analysis – Phase to design analysis model and system requirements analysis.

- Design – Phase to design the system and its components.
- Implementation – Phase to implement the develop system.
- Maintenance – Phase to provide support for any errors and upgrade the new features.

Abstractions: This is applicable breakpoint layers and traceability for enterprise architecture design to be followed in implementation. It refines the enterprise complexity that results to manage the risks could occur in enterprise systems that are directly concern the to stakeholders. This is the matrix for the different perspectives of enterprise frameworks.

- What – It is the composition of the objects used to build the system.
- How – It specifies the functional requirements, process and models for system.
- Where – This level defines the location of components for enterprise.
- Who – This defines the proper allocation of uses and models.
- When – It includes the management of lifecycle and different phases of development.
- Why - It includes the ends/means of the process, and analysis of enterprise system.

Non-functional requirements: They are the specific emergent characters or properties offered by systems that affect they system or performance of the whole system indirectly. They are difficult to verify but quantified. These are the features of the frameworks that should be measured to know if they are maturely develop to adapt changes, flexible requirements, defined models and views. There could be several non-functional requirements depending upon, what kind of attributes should be measured. Usually they are not defined as elements of framework, but describes what framework are in featured basis. All these non-functional requirements from Table 3 are described in section 6.3

6.3 Attributes for non-functional requirements

These are the specific objective to define certain atomic properties of each enterprise architecture framework. The non-functional requirements are hard to trace and particular difficult to verify their measurement unit. Correctness, consistency, traceability and requirement interaction management are the prime issues to be dealt [21]. They define the atomic properties of each features of framework. The following non-functional requirements are considered to measure the capabilities of the framework.

- Adaptability – It is the ability of a system to acquire the changes in requirements or can accept new requirements.

- Compatibility – It describes the facility of systems to exchange information without replacing the features from each other.

- Cohesiveness – It is the feature of module in system to execute the tasks effectively. Cohesion is the uniqueness in purpose of the system elements.

- Conceptuality – It represents domain under concept, which is the details of requirements to be manifested in code. This is the conceptual perspective of developers for the end users.

- Configurability – It describes the capacity manage the different components of the software. It is a software process to organize and control each items and information in software.

- Consistency – It refers to uniform approaches and techniques of system specification between different system's design and development.

- Coupling – It describes the ability to interact between modules and components of the system.

- Diversity – It describes difference of components, modules, data structures and data types used in system.

- Dependability – It refers to the degree of software to perform expected functions and services relying on other software components.

- Extensibility – It is ability in architectural, data and procedural design extended by adding variations in design and development of software system.

- Flexibility – It is one of the most required ability in development process to be able to modify operational program if there is known changes in the deployment environment.

- Interoperability – It is the ability to exchange the services offered within the agreed protocols between the participants.

- Maintainability – It is an effort required to scan and fix an error in system in a easy way even though requirements are changed.

- Maturity – It describes state of system to offer full-featured services and functionality at the state of development.

- Portability – It is the ability of the system to transpose from one to another environment without any disruptions.

- Robustness – It is the ability of a system to recover elegantly after failure or restart.

- Scalability – It refers to the ease of a system to be made smaller or larger to adjust the environment.

- Security – It is a mechanism to alert the possible threats to information and system process.

- Usability – It is the effort to use, handle and learn services or product functions in certain time.

This chapter concludes with the descriptions of attributes and perspectives/aspects to be measured for the comparison. The section 6.2 describes all the attributes required to build the enterprise system and 6.3 are the requirements that allows measuring features to build adaptive enterprise system.

Chapter 7 Enterprise architecture frameworks

7.1 Zachman Framework (ZF)

An enterprise architecture developed at IBM by John Zachman in 1987 [7] is one of the most acceptable frameworks in this domain named as "Zachman Framework" [7]. This provides the simplicity model of complex information systems implementations using logical architecture [10]. This is not just a methodology to process collecting, managing and utilization of refined information [11].

	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organisational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organisational Unit & Role Relationship Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role Relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location Details	Event Details

Figure 7.1: Zachman Framework (ZF)

The Zachman Framework is constructed around two-dimensional classification matrix that establishes a mutual language and set of perspectives for describing complicated enterprise systems. This has six communication questions: what, how, where who, when and why [12] in one dimension relationship of matrix with six perspectives: Planner, Owner, Designer, Builder, Subcontractor, and User. This focuses on establishing the views rather than process implementation and does not

have any explicit standard rules, however it assumes all the rules and relationships are followed and linked [12][13]. In the two-dimensional matrix, the rows define stakeholders and columns are the aspects in the architecture that follows the top-down approach of the row to align business concepts with actual physical process, riles and locations. The framework allows analysing the different people to look at same process from different six directions of six different transformations, which gives a holistic idea of system environment [14].

The framework is the set of rules of rows, columns and intersected cells in the matrix, which defines scope in each case in columns of matrix. Zachman suggested every architectural artefact should be once in a cell only, so that it can use the grid to refine the focus of each of all artefacts. Every cell in the grid should be populated with suitable artefacts to see complete view of the architecture of the system. Each cell in columns should be related to each other to give the meaning of full relationships in a row. As a whole framework, it seems to be the base for information system architecture which defines the context in different why, how, what who, where and when descriptions. This framework has long history and improvements along the time that keeps it existence in the industry till now. This framework is easier, fewer terms to define and gives the basic model of the system architecture.

7.2 The Open Group Architecture Framework (TOGAF)

The evolution of larger complex systems in industry made few other architecture discoveries like TOGAF, which was developed by The Open Group named as The Open Group Architecture Framework (TOGAF) [15]. The current version of TOGAF is 9.1, which is acceptable by different industries and popular for different kinds of domain. The flexibility and ability to solve the wide range of sub architectures makes it popular in various industrial system developments. This has very clear and strong definition of enterprise architecture process called Architecture Development method using open system building blocks as artefacts [1]. It has powerful strength for developing good principles than providing set of principles for system architecture. It divides EA in four categories of Business architecture that describes the business

goals, Application architecture to describe the interaction of each applications, Data architecture to store the data and policy to access the data, and Technical architecture for hardware and software infrastructure to support applications and their communications.

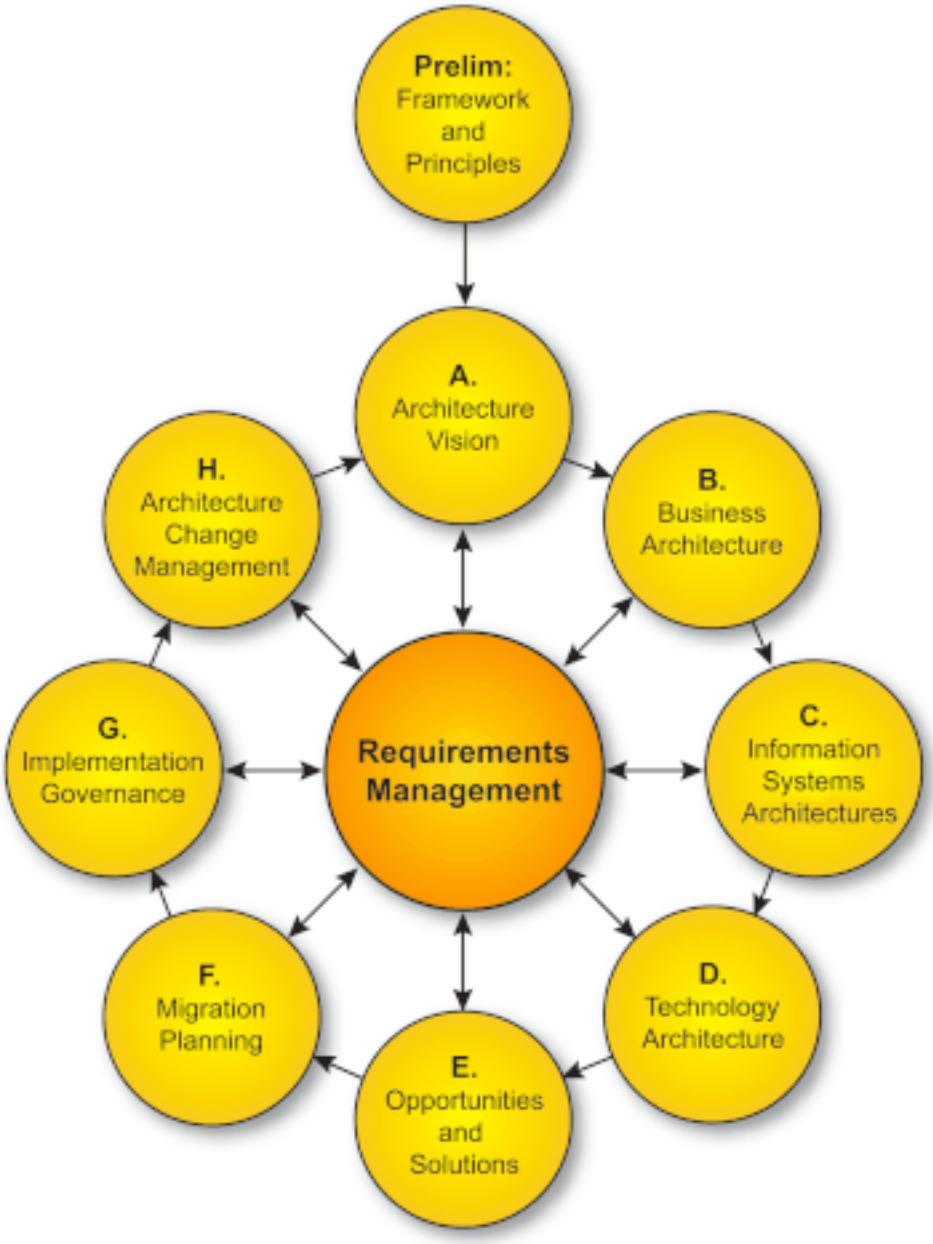

Figure 7.2: The Open Group Architecture Framework (TOGAF)

The most important core of TOGAF is ADM [15], which is overall architecture process to create artefacts. TOGAF defines enterprise continuum as the continuum of all the enterprise architectures of different ranges from highly generic to specific. All the generic architectures, which can be used in any domains in the universe, are defined as foundation architectures by TOGAF. Another level is common systems architectures, which can be seen in many domains but not in all enterprises. This is made specific with another level called industry architectures, which can be found in many enterprises of similar domain.

TOGAF has two knowledge bases, technical reference model (TRM) [15] to describe a generic IT architecture and standards information base (SIB) [15] to build IT architecture. As a whole, TOGAF is the rules for developing decent principles, rather than a set of architecture ethics. Providing guidance of IT resources, developing architecture principles and its implementation are three main levels of decision-making support across the entire enterprise architecture development.

7.3 Department of Defence Architecture Framework (DODAF)

This framework available in the defence domain is Department of Defence Architecture Framework, which is used by United States Department of Defence suitable for large systems with complicated integrations, and interoperability confronts [16]. This framework follows six steps of architecture development process that is data centric rather than focused in product. It provides the visual representation of data, information, and architecture description for decision makers. The development process starts with Step 1 to step 6 to define purpose, method, data, process and information of the architecture development process to document result in accordance with decision-maker needs: The final step of the architecture development process to create architectural meaningful presentation for decision markers.

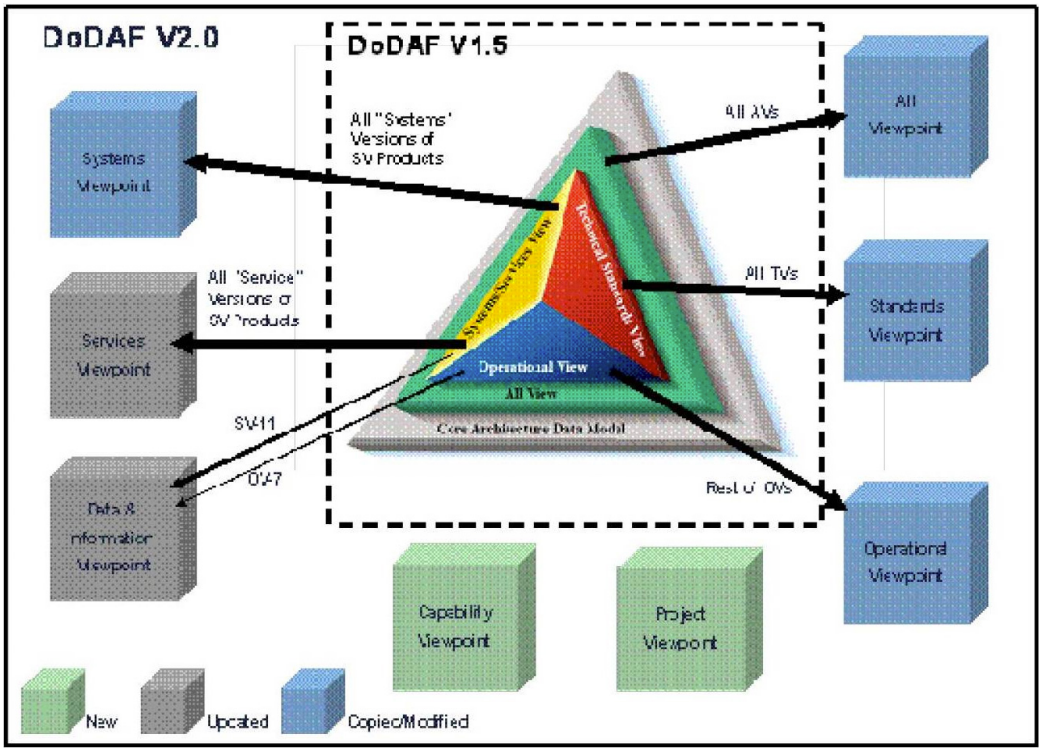

Figure 7.3: Department of Defence Architecture Framework (DODAF)

DoDAF uses Core Architecture Data Model (CADM) for architecture documentation. CADM is a standardized taxonomy to define views and their elements in a database. All Views provide an overview, summary and integrated dictionary of the architecture; Operational Views describe the business and operation of the architecture, they describe operation nodes, nodes connectivity, information exchange, organization relationship, operation rules, event-trace and logical data model; System Views describe the system and its components; Technical Views describes the current standard profile and future technical standards forecast [7].

It provides decoupling enterprise entities using only data layer and presentation layer that result in degrading dimensional attributes. It also provides scope integration concept using 'integrated architecture' supporting scope integration related attributes. Linkage architecture model and system life cycle model and transition plan are not specified within the framework.

7.4 Federal enterprise Architecture Framework (FEAF)

Federal enterprise Architecture Framework (FEAF) was developed and published by the US Federal Chief Information Officers (CIO) Council [17]. After the industry trend of developing enterprise architecture, government started to design own framework to guide large complex systems development. In 1996, FEAF was the response to the Clinger-Cohen Act, [17], which was used by Federal Agency CIOs to create, support, and accelerate integrated systems architectures.

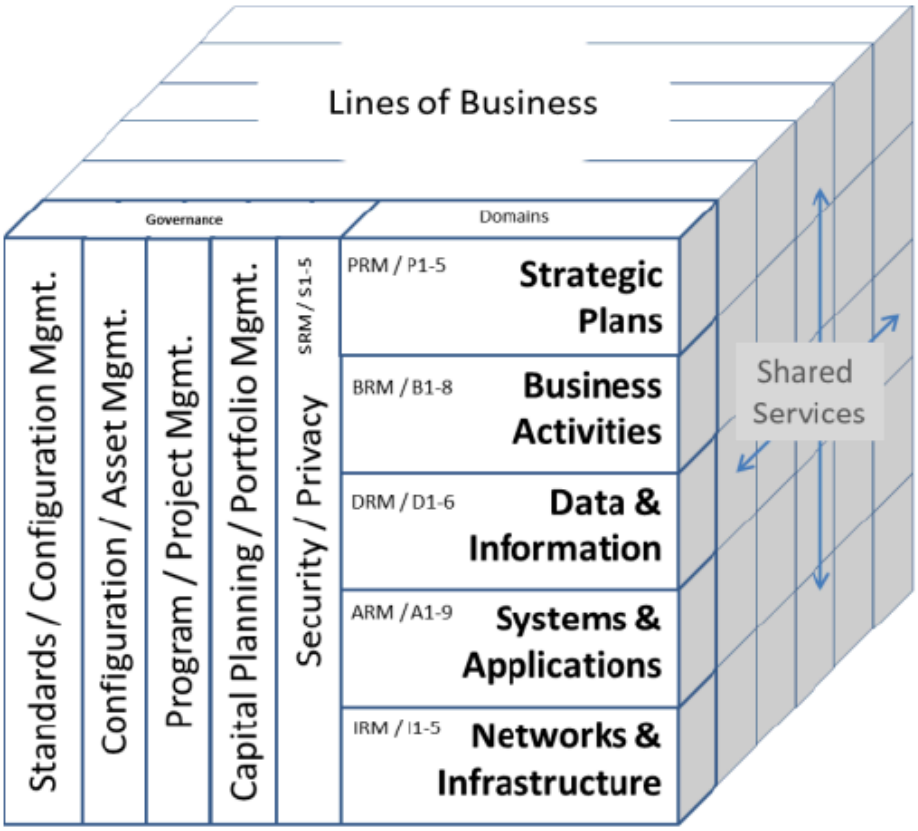

Figure 7.4: Federal enterprise Architecture Framework (FEAF)

The target for FEAF is to manage and share the Federal Information around the entire Federal Government [17]. It is the common language used by the agencies to share information of federal government. It is the most complete methodologies so far as it has comprehensive taxonomy like Zachman framework and the process of TOGAF. Its architectural sections are created separately but under structured guidelines, which

are considered as an enterprise within the Federal enterprise. FEAF consists of five reference models: performance reference model, business reference model, service component reference model, data reference model and technical reference model. FEAF is based on these reference models, which communicate with a common taxonomy and ontology for defining IT resources. In 2006, FEAF took different business perspective by defining three levels of architecture. Segment architecture level states a fundamental mission area, business services and enterprise goals. Lastly solution architecture level define IT assets like applications and component to improve agency business operations. Solution architecture is commonly related to segment architecture and enterprise architecture through definitions and constraints. FEAF is a Practical Guide [18] because it delivers supervision to U.S. federal agencies for frameworks. FEA lets flexibility in the practice of methods, work products, and tools to be used by the distinct federal agencies.

7.5 Gartner's enterprise Architecture Framework (GEAF)

This framework is developed by Gartner, who defined it as a verb, not a noun [30]. This EA is based on three constituents of business owners, information specialist and technology implementers. The common vision of these three building blocks of architecture is the key for successful business value that derives profitability [37]. The revised process model of Gartner EA in 2005 believes that the enterprise architecture should lead with future position of organization, rather than current position [30]. It puts more strength of single shared vision of the future, where the proposition of business, information, technical and solution architectures is shared. Gartner believes in the strategy of enterprise not about the engineering of it. Gartner enterprise Architecture Process: Evolution 2005 [30] presents the enterprise Architecture (EA) Process Model as a foundation, which provides a chromatic origin of thinking process in successful EA.

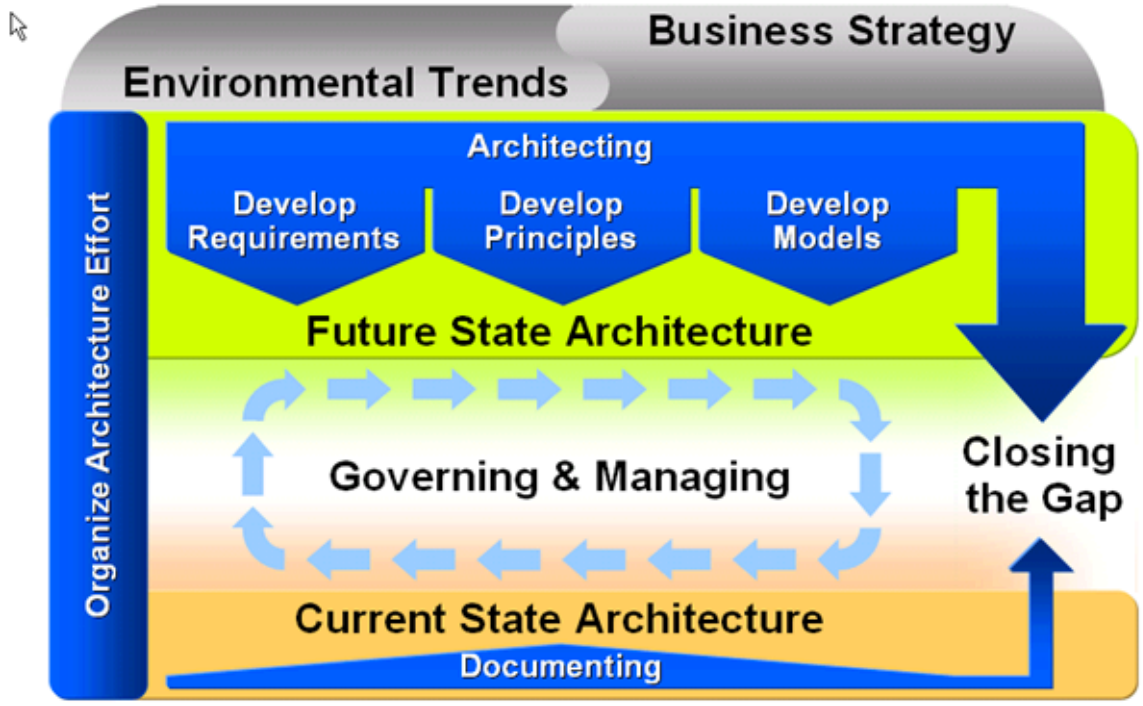

Figure 7.5: Gartner's enterprise Architecture Framework (GEAF)

The Gartner EA Process Model (see Figure 1) provides a logical approach to develop enterprise architecture for organization. It has multiple, iterative and nonlinear model for several sub processes. The initial stage of this EA was started with technical layer and later business architecture layer extended the technology border beyond. The process has gap analysis and portfolio management of future verses present. This revised process model shows two distinct states of future and current on the basis different EA building blocks. The building block of this process model consists of an enterprise view, environmental trends, business strategy, organized architecture effort, future-state architecture, Current-State Architecture – documenting, and finally the closing gap.

The Gartner EA Process Model is a precious base, trustworthy and vendor-neutral enterprise architecture framework that can add significant value to the architecture discipline in the development of enterprise architecture framework for any kind of system.

7.6 Treasury enterprise Architecture Framework (TEAF)

This framework was published by the US Department of the Treasury and published in July 2000 but later in May 2012, Federal enterprise Architecture Policy documented it in "The Common Approach to Federal enterprise Architecture"[17]. As mentioned in [38], the main purpose of its establishment is to manage the foundation structures and assets; identity and achieve objectives; and support mangers, business and technical planners.

TEAF has two dimensions for organization matrix, with each four views for vertical and horizontal perspectives. The structure of the frameworks has view containing 16 units and 4 perspectives, which is similar to the ZF columns and rows [38]. It divides the architecture in three parts views, perspectives and work products.

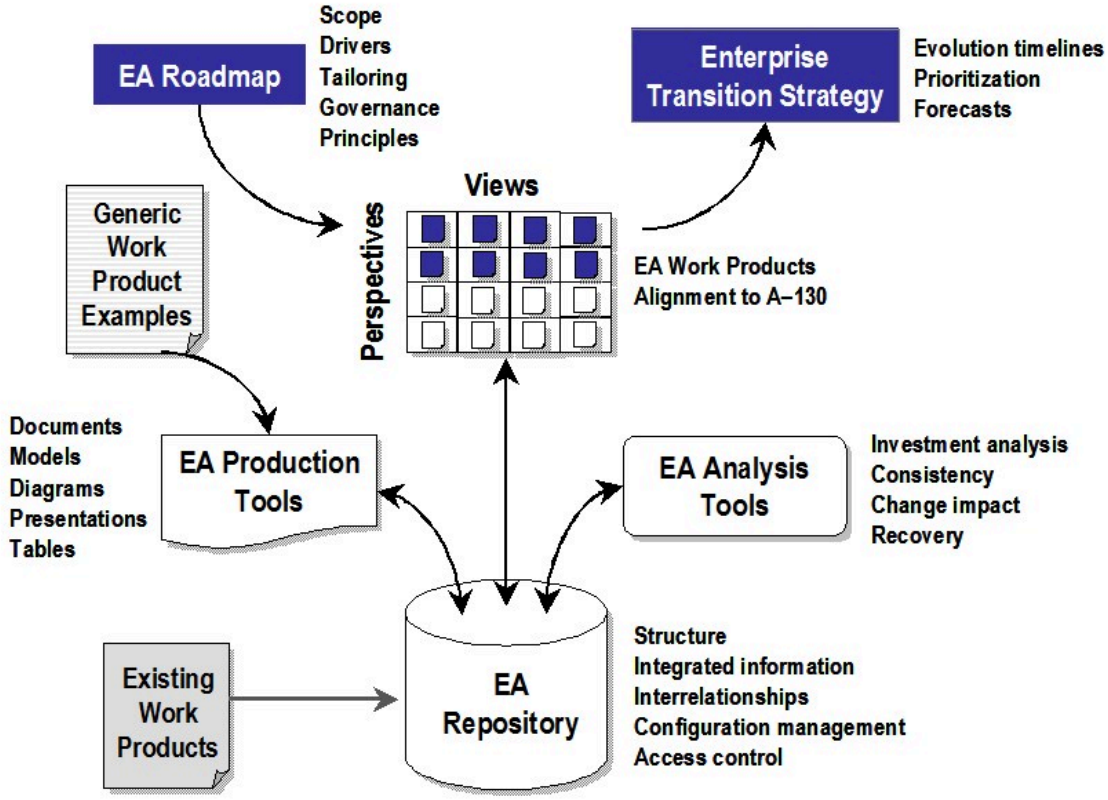

Figure 7.6: Treasury enterprise Architecture Framework (TEAF)

The TEAF identifies, as shown in the figure, resources and work products for the direction of enterprise architecture development, work products representing the enterprise architecture description, and work products documenting an enterprise architecture implementation. Its enterprise life cycle includes the management, business, and engineering process cycle that aligns its business and it activities. Artefacts in TEAF are divided into direction, description and accomplishment, whereas repositories are to store all the information [38]. The architecture development process of TEAF has nine steps: planning, analysis, design, implementation, project fusion, functions, assessments, organization and control, and technology advancements for step by step development process transition [38].

TEAF defines new perspectives on stakeholders, roadmaps, view and plans for transition and relates enterprise architecture lifecycle. It defines the roles and responsibilities, address information security and assurance. It also offers the principles, structures and repository for interchange of information [38].

7.7 Reference Model of Open Distributed Processing (RM-ODP)

The ISO (International Standards Organization) and ITU-T (International Telecommunication Union) worked together to define reference model to integrate a wide range of future ODP standards for distributed systems and maintain consistency among them [33]. The reference model (known as RM-ODP, Reference Model – Open Distributed Processing) provides the coordination framework for ODP standards, creating a heterogeneous infrastructure that supports integration of distribution, interworking and portability. The complex and large distributed systems need a specific structuring framework to manage it effectively. And the purpose of the RM-ODP is to define such a framework. This can be solved by "viewpoints" to describe the system and the "transparencies" to detect the specific problems in distributed system uniquely [24].

RM-ODP lies on four fundamental International Standards of overview, foundations, architecture and architectural semantics. Overview describes the ODP, scope and

terms used in enterprise architecture development. Foundation covers the concerns and aspects in distributed processing functions and systems. Architecture is characteristics possessed by distributed processing system, which recommends usage of for rational grouping of associated areas of the enterprise. And finally Architectural semantics take care of modelling with enough details of every concerned area.

The concept of RM-ODP is to specify distinct viewpoints into the specification of a given complex system. Each of these viewpoints fulfils an audience with interest in a particular set of aspects of the system. And every viewpoint has a viewpoint language to clarify the vocabulary and presentation for the audience [33]. There are five generic viewpoints in RM-ODP frameworks and they are enterprise viewpoints, information viewpoints, computational viewpoints, engineering viewpoints and technology viewpoints.

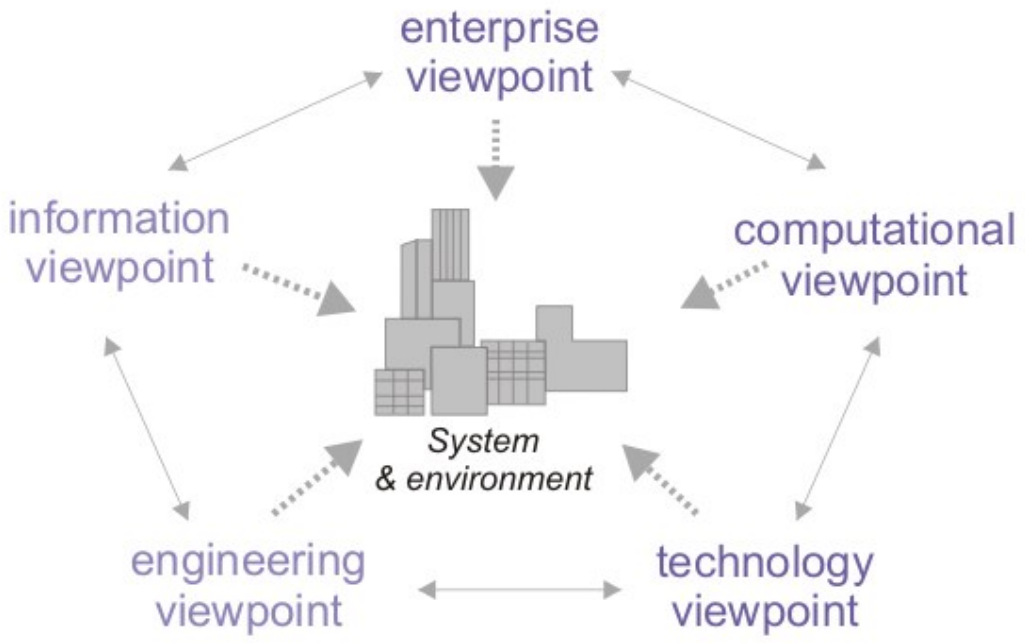

Figure 7.7: Reference Model of Open Distributed Processing (RM-ODP)

A viewpoint is a subdivision to certain specific area of concern throughout the design of the system. Viewpoints are not absolutely independent though they are specific to

different points. The key identifier is related to items in other viewpoints. RM-ODP ensures the mutual reliability among different viewpoints, which are bind together by a common object model. ODP systems specifications are defined in terms of interacting objects that contain information and provide services. This model has different level of distribution transparencies arising from number of concerns in distributed system. RM-ODP offers common functions, rules, conformance assessment and standards to define different functions in system like security issues.

7.8 British Ministry of Defence Architecture Framework (MODAF)

This is an enterprise architecture framework developed by the Ministry of Defence (MOD) to design defence planning and transformation of organisation activities. It is capable of rigorous information capture and keeps it coherent to understand complex issues in the organizational changes [32].

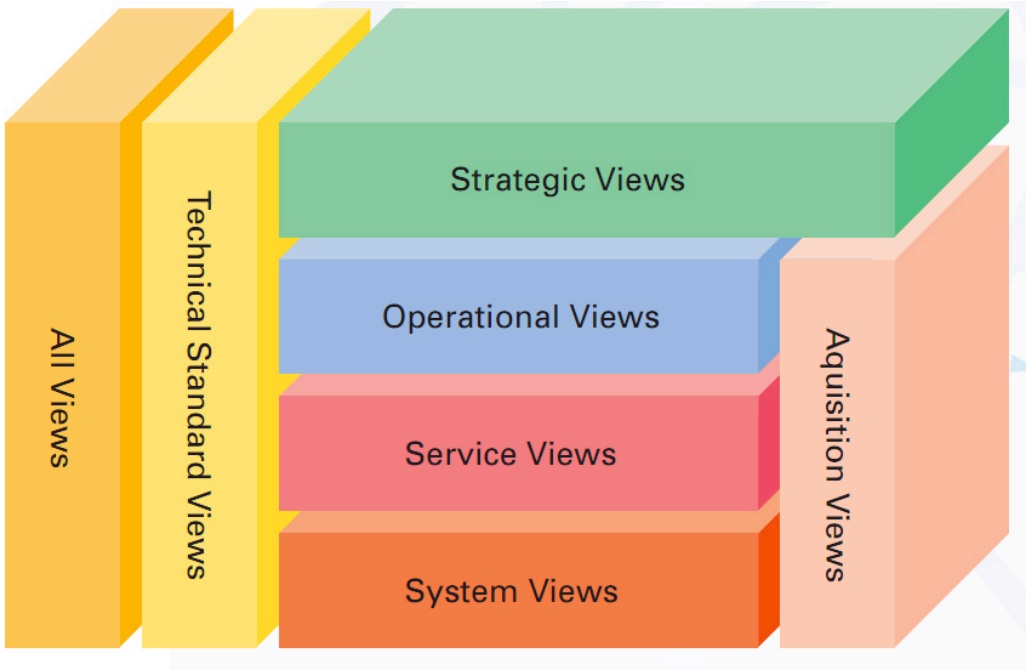

Figure 7.8: British Ministry of Defence Architecture Framework (MODAF)

MODAF is based on views, which are set of rules and models to define the relationships between data and views. As different stakeholders have distinctive

interest in a business, each view in MODAF allows viewing a business from different perspectives. The business area that is in investigation can be presented in graphical and textual visualisation with seven different coherent sets of rules offered by MODAF. Strategic views define the anticipated business result, and required potentials for its achievement. Operational views are the set of processes, information and entities required for the potential. The services required to support the process description in operational views in MODAF is described by Service oriented views. Systems views give solution by physical deployment of the operational and service orientated views. The dependencies and timelines to deliver the solution are mapped by Acquisition views. And lastly, Technical views describe the standards applied to the solution [32].

Meta model in MODAF provides a technical standard to allow data interchange between architectures produced in different modelling applications. This is called "M3" ontological data exchange mechanism. Although the MODAF Meta Model describes generic types of architectural information and their relationships, if re-use and integration of architectural products is required, those products must also utilise a common terminology and library of standard elements across architectures. This generic set of terminology and reference data in MODAF Ontology supports architectural coherence across the MOD, architectural comparison and data exchange clarity.

7.9 4+1 Architectural View Model

Philippe Kruchten designed the software-intensive systems architecture known as of 4+1 View Model for architectural analysis and modelling of the enterprise systems [24]. This framework describes large and challenging software architecture with the model of multiple views or perspectives. The model is made up of five main views as in the figure below:

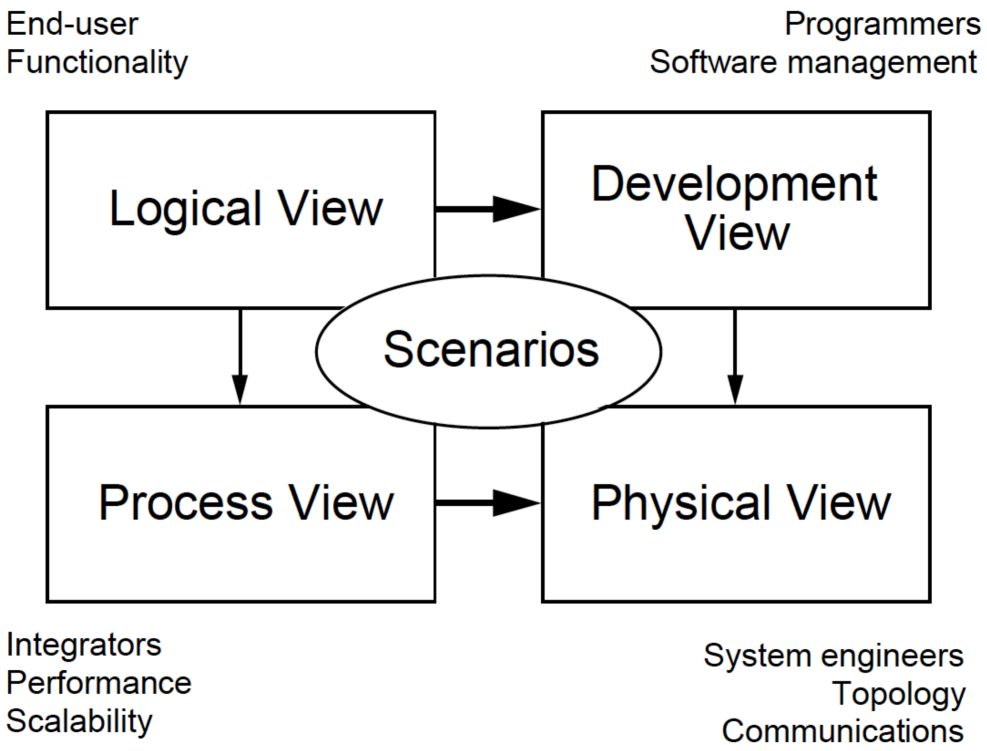

Figure 7.9: 4+1Architectural View Model

Starting with, the Logical View is for the functional requirements, followed by Process View to manage tasks, process, and their inter-communication in distributed environment including non-functional requirements. The Development View organizes software modules, and Physical View specifies the mapping of node from software to hardware.

The description and decisions are made based on these four views and the fifth view Scenario is the use cases use to discover and test the architecture [25]. This model follows an iterative approach for analysis and decomposition of the architecture design. The framework has powerful feature for development of distributed systems and uses UML notation for outcome model. The review by [25] mentions that the model does not care about surrounding environment, along with the unclear model for risk assessment and detail designs. The documentation by Philippe Kruchten [24] mentions the iteration phases as: sketching, organizing, specifying and optimizing validating for the architecture design. This allows the requirements to be polished, complete and better understood for stakeholder, engineers, end-users and consumers. The use of layers with interfaces, subsystems, modules is described by the term

"blueprints" in the whole development process, which is similar to the idea of artefacts.

7.10 Extended enterprise Architecture Framework (E2AF)

In 2002, the Institute for enterprise architecture developments created this framework. As mentioned by Jaap Schekkerman, it is a holistic method of four enterprise architecture components: business, information, technical and applications perspectives developing a streamlined grid of views that describe the concern areas within an organization [29]. This is a widely known communication Framework to define the topics and relations linked up between stakeholders during an architecture program. This also has strong emphasis on contextual responsiveness of threats and opportunities provide flexibility and adaptation to changing business environments.

The six levels to define the four components in this framework are as: Contextual level, Environmental level, Conceptual level, Logical level, Physical level and Transformational level.

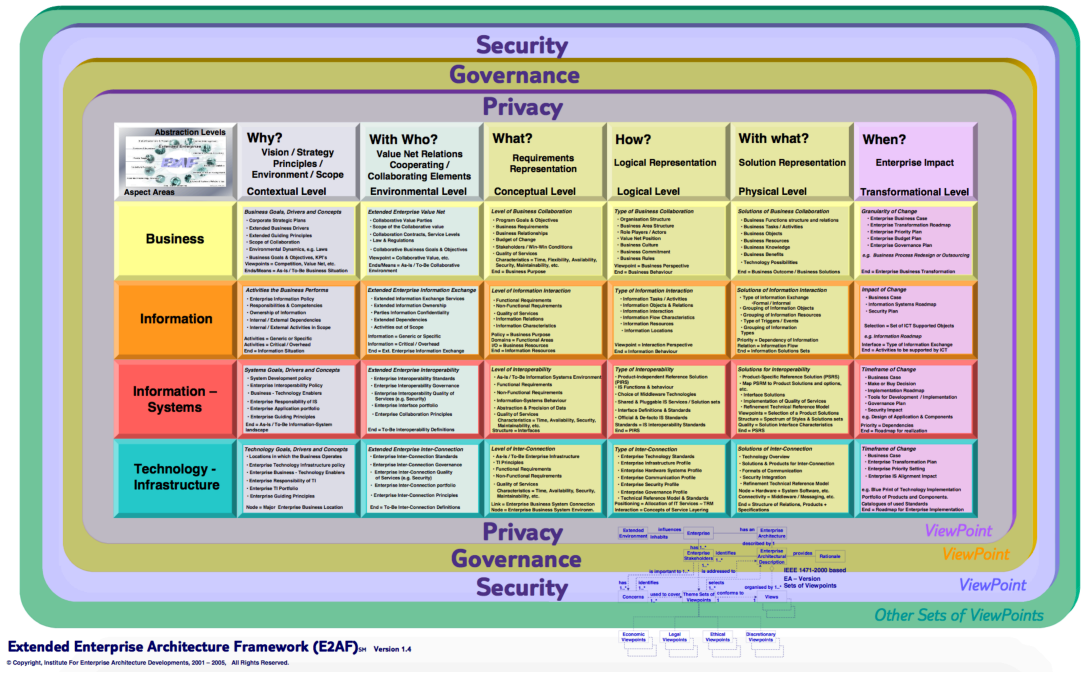

Figure 7.10: Extended enterprise Architecture Framework (E2AF)

As a whole, The Extended enterprise Architecture Framework offers a flexible model with emphasis in communication structures of organization to adapt the agile business needs and relationships amongst stakeholders [39].

7.11 SABSA (Sherwood Applied Business Security Architecture)

This is an open source methodology developed for designing secure architectures addressing information security, assurance and deliverable solutions in critical business. This model is completely vendor-neutral, scalable and fills the gap for security architecture and security service management [19].

Regarding security issues in architecture design, SABSA developers believe that the secure strategies are supportive to long-term cost analysis and business [23]. SABSA adopts a Business-Driven method for EISA development that describes the relationship of technical and procedural solutions and it supports the interoperability as a major business requirement for business criteria [23].

	ASSETS (What)	MOTIVATION (Why)	PROCESS (How)	PEOPLE (Who)	LOCATION (Where)	TIME (When)
CONTEXTUAL ARCHITECURE	Business Decisions	Business Risk	Business Processes	Business Governance	Business Geography	Business Time Dependence
	Taxonomy of Business Assets, including Goals & Objectives	Opportunities & Threats Inventory	Inventory of Operational Processes	Organisational Structure & the Extended Enterprise	Inventory of Buildings, Sites, Territories, Jurisdictions, etc.	Time dependencies of business objectives
CONCEPTUAL ARCHITECTURE	Business Knowledge & Risk Strategy	Risk Management Objectives	Strategies for Process Assurance	Roles & Responsibilities	Domain Framework	Time Management Framework
	Business Attributes Profile	Enablement & Control Objectives; Policy Architecture	Process Mapping Framework; Architectural Strategies for ICT	Owners, Custodians and Users; Service Providers & Customers	Security Domain Concepts & Framework	Through-Life Risk Management Framework
LOGICAL ARCHITECTURE	Information Assets	Risk Management Policies	Process Maps & Services	Entity & Trust Framework	Domain Maps	Calendar & Timetable
	Inventory of Information Assets	Domain Policies	Information Flows; Functional Transformations; Service Oriented Architecture	Entity Schema; Trust Models; Privilege Profiles	Domain Definitions; Inter-domain associations & interactions	Start Times, Lifetimes & Deadlines
PHYSICAL ARCHITECTURE	Data Assets	Risk Management Practices	Process Mechanisms	Human Interface	ICT Infrastructure	Processing Schedule
	Data Dictionary & Data Inventory	Risk Management Rules & Procedures	Applications; Middleware; Systems; Security Mechanisms	User Interface to ICT Systems; Access Control Systems	Host Platforms, Layout & Networks	Timing & Sequencing of Processes and Sessions
COMPONENT ARCHITECTURE	ICT Components	Risk Management Tools & Standards	Process Tools & Standards	Personnel Man'ment Tools & Standards	Locator Tools & Standards	Step Timing & Sequencing Tools
	ICT Products, including Data Repositories and Processors	Risk Analysis Tools; Risk Registers; Risk Monitoring and Reporting Tools	Tools and Protocols for Process Delivery	Identities; Job Descriptions; Roles; Functions; Actions & Access Control Lists	Nodes, Addresses and other Locators	Time Schedules; Clocks, Timers & Interrupts
SERVICE MANAGEMENT ARCHITECTURE	Service Delivery Management	Operational Risk Management	Process Delivery Management	Personnel Management	Management of Environment	Time & Performance Management
	Assurance of Operational Continuity & Excellence	Risk Assessment; Risk Monitoring & Reporting; Risk Treatment	Management & Support of Systems, Applications & Services	Account Provisioning; User Support Management	Management of Buildings, Sites, Platforms & Networks	Management of Calendar and Timetable

Figure 7.11: SABSA (Sherwood Applied Business Security Architecture)

The SABSA Model is six-layered architecture as in figure above. It follows the developing model for enterprise architecture, with modified security view. Each layer describes the view of different perspective for design, construct and use of building blocks. The contextual layer describes the business driver, risk, management, relationships, point-of supply management and performance management.

Sherwood, Clark and Lynas [19] have mentioned SABSA as a combination of best practice and its compliance with security standards. The matrix of six layers is described as each perspective of Business View, Architect's View, Designer's View, Builder's View, Tradesman's View and Service Manager' View.

This model provides life cycle for the development and risk management processes from designer's initial vision to specialist's components. The method of requirement engineering is specific to SABSA that establishes relation of business policy and technical resolutions [23]. Business Attributes Profile in SABSA is the core factor, as it defines business requirements and gathers the required guidelines. Another major factor in this model is the account of time dimension for requirement gathering,

which emphasises the security of information in the continuous process. Brian Ritchot [40] mentions SABSA as a powerful framework with essential background theory to deliver the business security architecture.

7.12 The Oracle enterprise Architecture Framework

Oracle took initiation to develop their own efficient and business-driven hybrid framework inspired by TOGAF, FEAF and Gartner. The Oracle enterprise Architecture Framework (OEAF) is simple, practical and prescriptive to help customers align their IT and business strategies [4]. The main purpose of this framework is to provide "just enough" structure that can be developed "just in time" business requirements. This provides well structure architecture to share intellectual capital with its customers and partners, increasing Oracle's strategic business value proposition.

The OEAF supports agile capabilities for mapping business requirements to deployment. The approach attempts to exclude complications and unnecessary buildings of other frameworks to quick and incremental results in an enterprise. The processes and artefacts are reduced to appropriate level to meet the business objective. This model follows the simultaneous development of the several modules and avoids waterfall process. In addition, it includes the prebuilt reference architectures from logical to physical components to minimize implementation risks.

There are seven components in OEAF to build enterprise architecture. Beginning with Business architecture to define business strategy, functions and organizational structure followed by application architecture that ties business functions to application strategy, services, and components.

The information architecture describes the components for managing information and its proper sharing. And the last, technology architecture describes the infrastructures for strategy, services, logical, and physical components.

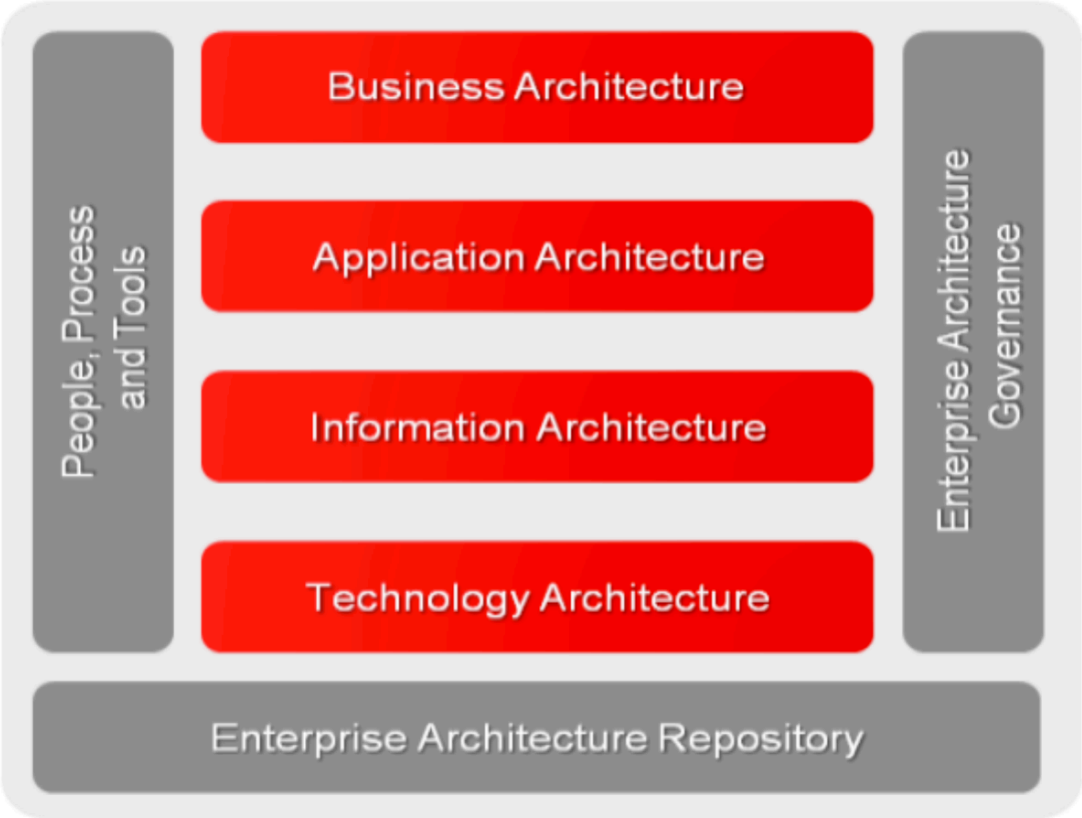

Figure 7.12: Oracle Enterprise Architecture Framework (OEAF)

The architecture identifies people, processes and tools for developing and managing enterprise architecture. The EA repository in OEAF is responsible for all the architecture artefacts and deliverables during the development process. The Oracle Architecture Development Process (OADP) defines an applied and collective approach to align the enterprise and solution architecture with their business goals.

7.13 Generalised enterprise Reference Architecture and Methodology (GERAM)

This method is carried out by AMICE Consortium previously to provide guidelines for enterprise integration programs but later IFIP/IFAC task force recognised it to define a generalized architecture to satisfy service demands and business needs [41]. GERAM is all about methods, models and tools to build the integrated enterprise with

GERA life cycle, which has different phases. The phases in GERAM are broken into activities to identify entities and tasks to manage the operational effort, which replace the traditional processes.

Modelling Framework of GERA provides analysis and modelling lifecycle approach to define scope and content of enterprise from different dimensions. Life-cycle dimension provides modelling process and activities. Genericity dimension controls the process from generic and partial to specific. View dimension provides specific view of the enterprise entity.

There are four entity views in GERA modelling framework, which are Entity Model Content Views, The Entity Purpose Views, The Entity Implementation View and Physical Manifestation Views to describe all the Software views and hardware views.

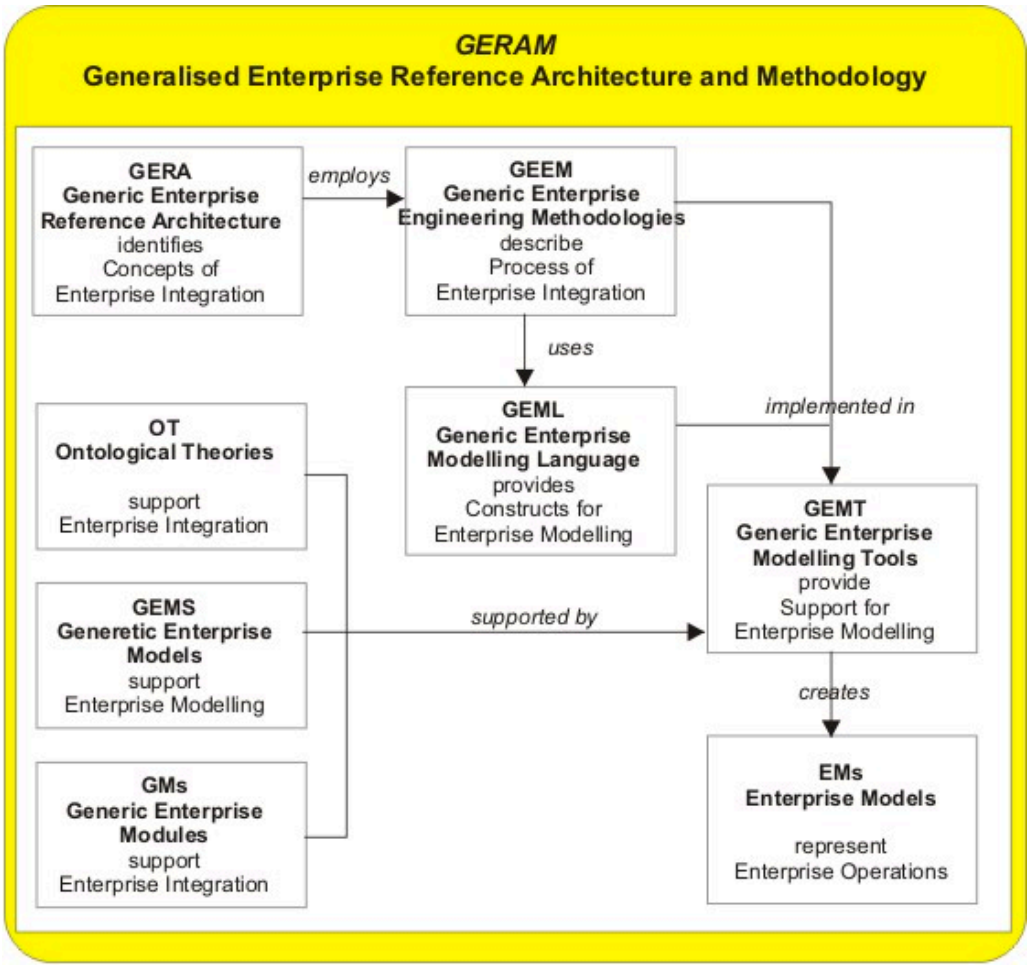

Figure 7.13: Generalised enterprise Reference Architecture and Methodology (GERAM)

It offers the enterprise engineering tools to deploy enterprise-modelling languages in support of methodologies, and manage enterprise models. It supports the analysis and evaluation models to help in decision-making process in the progress of enterprise engineering.

The continuously maintained model of particular enterprise entity represents requirements of the user and application allowing interoperability among the models of other enterprises. Enterprise models creation and use are conveyed in enterprise-modelling languages in the real-time for serving real time services [41].

7.14 Model Driven Architecture (MDA)

This model is an open, vendor-neutral approach built by Object Management Group (OMG) to challenge the business and technology change. It offers the standards and separates the business and application layers from independent platform technology [41][34]. As mentioned by Leist, and Zellner "The MDA defines an approach to IT system specification that separates the specification of system functionality from the specification of the implementation of that functionality on a specific technology platform" [34].

This model offers the procedure model for development process with its own specifications documents described by UML models. It consists four steps starting with creating a computation independent model (CIM) followed by creating a platform independent model (PIM). The third step is creating platform specific model (PSM) with final step for generating the application [34]. The meta-models in MDA are expressed with a specific abstract syntax in UML or Meta-Object Facility or any other languages [34]. The viewpoint specifies models elements for particular stakeholders expressed through Meta model elements. Each View conforms to a viewpoint of a system [41].

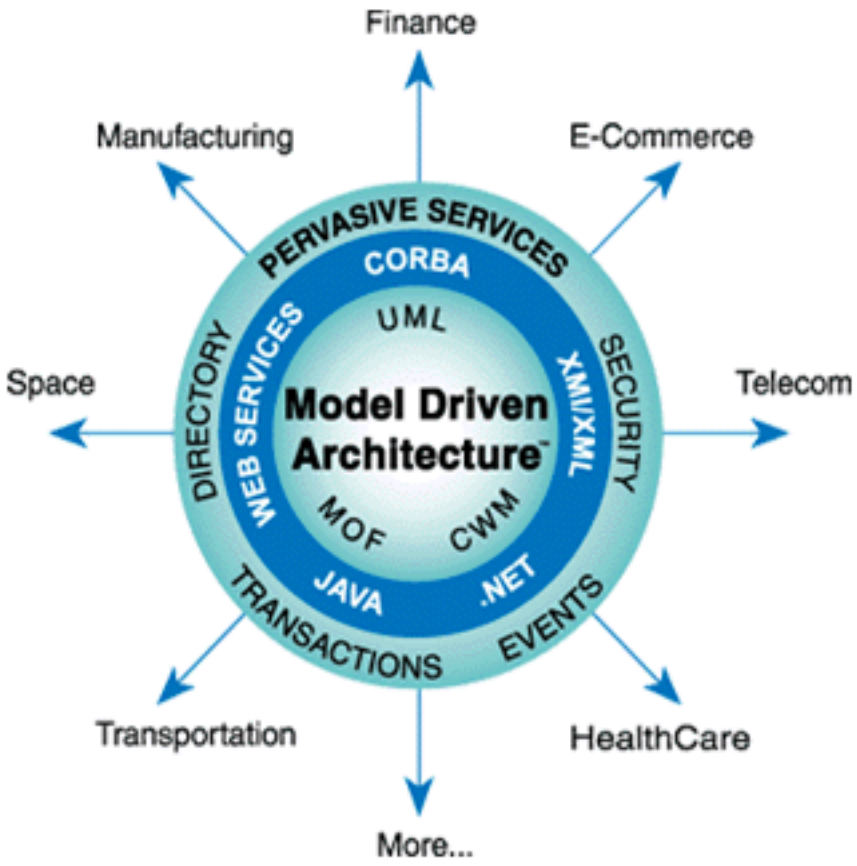

Figure 7.14: Model Driven Architecture

Artefacts in MDA are the portion of evidence in development process of systems likes modelling files, sourcing files etc. The roles in MDA do not describe details although the business and modelling experts work together with business functionality and behaviours. MDA tools are used to process the models and Meta models for different purposes like creating, analysis, transformation, composition, testing, simulation, management, and engineering.

Chapter 8 Assessments and evaluation of enterprise architecture frameworks

8.1 Assessments

Comparison assessments for all attributes of enterprise architecture frameworks are carried out based on chapter 7 for each perspectives/aspects. This chapter shows all the assessment views, domains, goals, inputs, outputs, guide, artefacts, tools, system development lifecycle, abstraction, and non-functional requirements. The measurement scale for the assessment is as mentioned in Chapter 6.

Fully Support = 2

Partial Support = 1

Does not support/Not Mentioned = 0

8.1.1 Fundamental elements

This is the general finding for all the attributes from each enterprise architecture framework. This shows the overall general comparison for the attributes listed in Chapter 6. This table values are exclude for the total results. This comparison is to find fundamental elements of enterprise frameworks.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Views	2	1	1	1	0	2	2	2	2	2	2	2	2	1
Methods	1	2	1	0	0	0	0	0	1	0	1	0	0	2
Processes	2	2	1	1	2	1	0	1	2	2	2	2	2	2
Domains	1	2	1	1	2	1	1	0	0	1	2	2	1	2
Roles	0	0	1	2	0	2	0	0	0	2	2	1	2	2
Artefacts	2	2	2	2	2	2	1	0	0	0	0	2	0	2
Tools	2	2	1	2	2	2	2	2	1	2	2	2	1	2
System														

Abstraction	2	1	2	2	2	2	2	2	2	2	2	1	1	2
Total	**14**	**14**	**10**	**13**	**12**	**14**	**8**	**9**	**8**	**13**	**13**	**14**	**9**	**17**

Table 4: Comparison of fundamental elements

There general comparison shows MDA offers all the attributes, followed by OEAF, TEAF, TOGAF and ZF. This is only overall view, which has to undergo through each of the attributes and their aspects to major its depths. The following sections from 9.3 to 9.11 are the assessment for each of these attributes.

8.1.2 Views

In this section, the table compares the Views of each frameworks and the most preferable is ZF, RM-ODP, MODAF, E2AF and SABSA.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Planner	2	0	2	2	0	2	2	2	2	2	2	0	2	0
Owner	2	2	2	2	0	2	2	2	2	2	2	0	2	0
Designer	2	2	2	2	0	2	2	2	2	2	2	0	2	2

Builder	2	0	2	2	0	2	2	2	2	2	2	0	2	2
Subcontractor	2	0	0	2	0	0	2	2	0	2	2	0	0	0
User	2	0	0	0	0	0	2	2	2	2	2	0	0	2
Total	**12**	**4**	**8**	**10**	**0**	**8**	**12**	**12**	**10**	**12**	**12**	**0**	**8**	**6**

Table 5: Comparison based on views

8.1.3 Domains

This shows the Domains offered by different frameworks, where ZF, TOGAF, FEAF, RM-ODP, E2AF and OEAF are the most preferable.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Business Architecture	2	2	0	2	2	0	2	2	0	2	2	2	2	0
Data Architecture	2	2	2	2	2	2	2	0	0	2	2	2	2	2

Application Architecture	2	2	0	2	1	2	2	2	0	2	0	2	0	1
Technology Architecture	2	2	0	2	2	2	2	2	0	2	0	2	2	2
Total	**8**	**8**	**2**	**8**	**7**	**6**	**8**	**6**	**0**	**8**	**4**	**8**	**6**	**5**

Table 6: Comparison based on domains

8.1.4 Processes

This attribute has other three different sub attributes: Goals, Inputs and Outputs from each framework. Comparing goals attributes shows MDA is the most preferable framework followed by MODAF and TOGAF. In comparison of Inputs TOGAF, SABSA and OEAF are the equally preferable followed by DODAF, FEAF, MODAF and E2AF. Outputs show OEAF is the most desirable followed by TOGAF and MDA.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
	Goals													
Architecture Definition and	1	2	2	2	2	2	2	2	1	2	2	2	2	2

understanding														
Architecture development process	1	2	2	2	1	2	0	2	0	2	2	2	2	2
Architecture evolution	0	2	2	2	2	2	1	2	0	2	2	2	2	2
Architecture analysis	2	2	2	2	1	1	2	2	2	1	1	1	2	2
Architecture models	2	2	2	2	2	2	2	2	2	2	2	2	1	1
Architecture knowledge base	0	2	0	2	0	2	1	2	0	1	1	1	2	2
Architecture verifiability	0	2	2	0	0	0	1	2	2	2	2	1	2	2
Design trade-offs	1	1	2	1	1	1	1	2	2	2	1	1	1	2
Design rationale	1	2	1	1	1	1	2	1	2	2	1	2	2	2
Standardization	0	2	2	2	2	1	2	2	2	2	1	2	2	2
Total	**8**	**19**	**17**	**16**	**12**	**14**	**14**	**19**	**13**	**18**	**15**	**17**	**18**	**21**

	Inputs													
Business drivers	1	2	2	2	2	2	1	2	1	1	2	2	1	2
Technology inputs	0	2	2	2	0	2	1	1	0	2	2	2	1	2
Business requirements	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Information systems	1	2	2	2	2	2	2	2	1	2	2	2	2	2
Existing architecture	1	2	2	2	2	2	2	2	2	2	2	2	2	2
Non-functional requirements	1	2	1	1	2	0	2	2	2	2	2	2	2	2
Total	**6**	**12**	**11**	**11**	**10**	**10**	**10**	**11**	**8**	**11**	**12**	**12**	**10**	**12**
	Outputs													
Business model support	2	2	2	2	2	2	2	1	1	2	2	2	1	2
System model support	2	2	2	2	0	4	2	2	2	2	2	2	2	2
Information model	2	2	2	2	2	2	2	2	2	2	2	2	1	2

support														
Computation model support	2	2	2	2	2	2	2	1	2	0	0	2	2	2
Software configuration	0	2	0	0	1	0	1	0	2	0	0	2	1	2
Software process incorporation	2	2	2	2	2	1	2	2	2	2	2	2	1	2
Implementation model	1	2	2	1	0	1	2	0	1	2	2	2	2	1
Platform	2	2	2	2	2	2	2	2	1	2	2	2	2	2
Non-functional requirement design	1	2	1	1	1	1	2	2	2	2	2	2	2	2
Transitional design	0	2	2	2	2	2	0	2	0	1	0	2	1	2
Design rationale	0	1	1	1	0	0	1	1	1	2	2	2	2	2
Total	**14**	**21**	**18**	**17**	**14**	**15**	**18**	**15**	**16**	**17**	**16**	**22**	**17**	**21**

Table 7: Comparison based on goals, inputs and outputs

8.1.5 Guide

In comparison of Guide attributes, MDA offers most of the aspects followed by OEAF, SABSA and E2AF.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Meta model	1	0	1	0	0	1	2	2	0	0	2	2	2	2
Procedure model	1	2	2	2	2	2	1	2	2	1	2	1	2	2
Modelling techniques	0	1	2	0	0	1	2	2	0	2	1	2	2	2
Role	1	0	1	2	2	2	2	2	0	2	2	2	2	2
Specific document	0	1	2	2	2	2	1	0	0	0	2	2	0	2
Taxonomy completeness	2	1	2	1	1	1	1	1	1	2	2	2	1	1
Process completeness	1	2	1	1	2	1	1	2	2	2	2	2	2	2
Maturity model	0	1	1	1	2	1	0	0	1	2	2	2	2	2

Reference model guidance	0	1	2	2	1	1	2	0	1	2	1	2	2	2
Practice guidance	0	1	2	1	2	1	1	2	0	2	1	2	1	2
Governance guidance	1	1	0	1	2	1	0	2	0	2	2	2	0	2
Partitioning guidance	1	1	1	2	1	1	2	1	0	1	1	1	1	2
Prescriptive catalogue	1	1	1	2	1	2	2	0	0	2	1	2	1	2
Vendor neutrality	0	2	0	1	2	1	2	0	2	0	2	2	0	2
Information availability	1	2	2	1	2	1	1	2	1	2	1	2	1	2
Time to value	1	1	1	2	2	1	1	1	1	1	1	1	1	1
Transformation	1	2	2	2	2	2	2	2	1	2	2	2	2	2
Total	**12**	**20**	**23**	**23**	**26**	**22**	**23**	**21**	**12**	**27**	**27**	**31**	**22**	**32**

Table 8: Comparison based on guide offered

8.1.6 Artefacts

Artefacts are mostly offered by FEAF, TEAF and OEAF followed by TOGAF.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Strategy	2	1	2	2	2	2	1	0	0	0	0	2	0	2
Process	2	2	2	2	2	2	0	2	2	2	2	2	2	2
Application	2	2	1	2	1	2	0	0	0	0	0	2	0	0
Software	0	2	1	2	1	2	0	0	0	0	0	2	0	2
Technical	0	2	2	2	2	2	2	0	0	0	0	2	0	2
Total	**6**	**9**	**8**	**10**	**8**	**10**	**3**	**2**	**2**	**2**	**3**	**10**	**2**	**8**

Table 9: Comparison based on artefacts

8.1.7 Tools

Tools are mostly offered by TOGAF, DODAF, FEAF, TEAF, and MODAF to design enterprise system.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Business process modelling tools	2	2	2	2	2	2	1	2	2	2	2	2	2	2
Data modelling tools	1	2	2	2	2	2	0	2	0	0	2	2	2	2
Code development tools	2	2	2	2	2	2	2	2	2	0	0	0	2	2
Network design and performance	1	2	2	2	0	2	2	2	1	0	0	0	0	1
Total	**6**	**8**	**8**	**8**	**6**	**8**	**5**	**8**	**5**	**2**	**4**	**4**	**6**	**7**

Table 10: Comparison based on development tools

8.1.8 System development lifecycle

Comparison of system development lifecycle attributes shows ZF, FEAF and GERAM follows the most perfect development cycle followed by E2AF, SABSA and OEAF.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Planning	2	0	2	2	0	2	0	2	2	0	2	2	2	2
Analysis	2	1	2	2	0	1	0	2	2	0	2	2	2	2
Design	2	1	2	2	0	2	0	2	2	0	2	2	2	2
Implementation	2	1	1	2	1	2	1	2	2	1	2	2	2	2
Maintenance	2	1	0	2	0	0	0	0	0	0	1	1	2	0
Total	**10**	**4**	**7**	**10**	**1**	**7**	**1**	**8**	**8**	**9**	**9**	**9**	**10**	**8**

Table 11: Comparison based on system development lifecycle

8.1.9 Abstraction

The comparison of all aspects in abstraction shows that ZF, E2AF, SABSA provides all level of abstractions.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
What	2	0	2	2	2	2	2	2	2	2	2	1	2	2
How	2	1	2	2	2	2	2	2	2	2	2	0	0	2
Where	2	0	2	2	0	2	0	2	0	2	2	0	2	2
Who	2	1	0	2	0	0	2	2	0	2	2	0	0	2
When	2	0	0	0	0	0	0	0	2	2	2	0	0	2
Why	2	0	2	0	2	2	2	2	2	2	2	0	0	0
Total	**12**	**2**	**8**	**8**	**6**	**8**	**8**	**10**	**8**	**12**	**12**	**1**	**4**	**10**

Table 12: Comparison based on abstraction

8.1.10 Non-functional requirements

The comparison of non-functional requirements verifies TOGAF offers the most extensive features for development of reference model for Trust Framework, which is followed by other frameworks like MDA, OEAF, and GERAM.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Adaptability	2	2	2	1	2	1	2	2	1	2	2	2	2	2
Compatibility	1	2	1	1	2	1	2	0	0	0	2	2	2	2
Cohesiveness	1	2	1	2	2	1	1	2	1	1	1	1	2	2
Conceptuality	1	2	2	2	2	2	2	2	2	1	2	1	2	2
Configurability	0	2	2	2	2	2	2	2	1	2	2	2	2	2
Consistency	1	2	2	2	2	2	2	2	2	2	2	2	2	2
Coupling	1	2	2	2	1	1	1	2	0	1	1	2	2	2
Diversity	1	2	1	1	2	1	1	1	0	1	1	2	1	2
Dependability	1	2	1	2	1	1	1	2	1	1	2	2	2	2

Extensibility	1	2	2	2	1	1	2	1	1	2	2	2	2	2
Flexibility	1	2	2	1	1	1	1	2	2	2	2	2	2	2
Interoperability	1	2	2	1	1	1	1	1	2	2	1	2	1	2
Maintainability	1	2	2	2	1	2	2	2	1	2	1	2	1	2
Maturity	1	2	2	1	2	1	1	2	1	2	2	1	2	2
Portability	1	2	1	1	2	1	2	2	1	1	1	1	1	2
Robustness	1	2	2	2	2	2	2	1	2	1	2	2	2	2
Scalability	1	2	2	2	2	2	2	2	2	2	2	2	2	2
Security	0	2	2	2	2	2	2	2	2	2	2	2	2	2
Usability	2	2	2	2	2	2	1	2	2	1	2	2	2	2
Total	**19**	**38**	**33**	**31**	**32**	**27**	**30**	**32**	**24**	**28**	**32**	**34**	**34**	**38**

Table 13: Comparison based on non-functional requirements

8.1.11 Conclusion

The final result shows that, MDA is the most advanced framework that offers all kind of features, attributes and methods required for the building reference model. This conclusion is followed by OEAF and TOGAF. The graph in below figure 16 shows the clear view of final result of comparison.

Review Parameters	ZF	TOGAF	DODAF	FEAF	GEAF	TEAF	RM-ODP	MODAF	4+1	E2AF	SABSA	OEAF	GERAM	MDA
Views	12	4	8	10	0	8	12	12	10	12	12	0	8	6
Domains	8	8	2	8	7	6	8	6	0	8	4	8	6	5
Goals	8	19	17	16	12	14	12	19	13	18	15	17	18	21
Inputs	6	12	11	11	10	10	10	11	8	11	12	12	10	12
Outcomes	14	21	18	17	14	15	18	15	16	17	16	22	17	21
Guide	12	20	23	23	26	22	23	21	12	27	27	30	22	32
Artefacts	6	9	8	10	8	10	3	2	2	2	2	10	2	8
Tools	6	8	8	8	6	8	5	8	5	2	4	4	6	7

System development lifecycle	10	4	7	10	1	7	1	8	8	1	9	9	10	8
Abstraction	12	2	8	8	6	8	8	10	8	12	12	1	4	10
Non-functional requirements	19	38	33	31	32	27	30	32	24	28	32	34	34	38
Total	**113**	**145**	**143**	**144**	**122**	**127**	**130**	**136**	**106**	**138**	**144**	**147**	**137**	**168**

Table 14: Final results of comparison

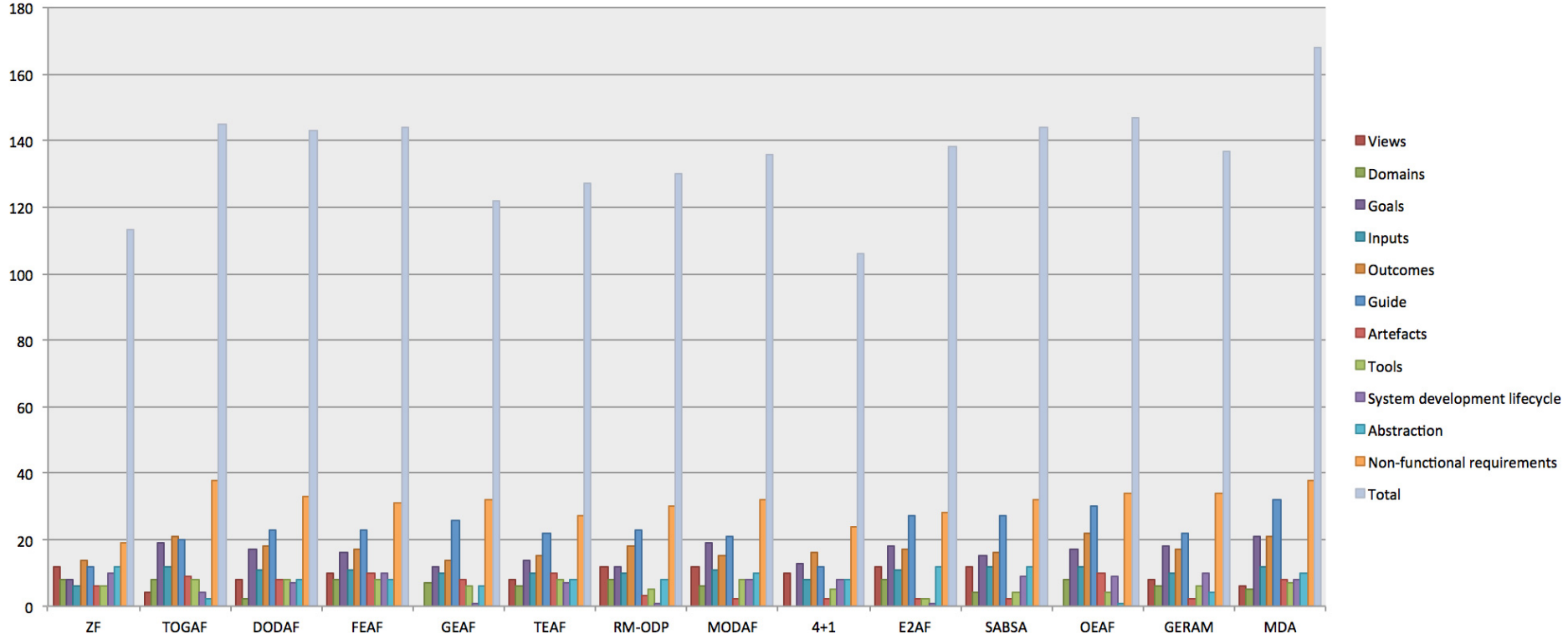

Figure 15: Graph for the final result comparison

8.2 Evaluation

This chapter is focused on the evaluation of the assessment performed in chapter 8. The result for comparison shows that MDA is the most closely and advanced framework that can be used to develop the Reference Model for Trust Frameworks. Although there are some limitations in each framework, which makes the selection process very difficult to decide the required solution. There were functional and non-functional requirements compared to measure the capability of the frameworks. Selection of enterprise architecture frameworks always depends on the domain of organization and range of problems and issue that should be solved by framework. The process depends on the business requirements, organizational portfolio and its business process modelling for its goals and mission. It is indeed significantly difficult to decide on choosing a single perfect method, as the changes in requirements and scenario of system implementation are always unique and constrained by time, scope and cost. Here the assessment uses the measurement process of Chapter 5 to decide that MDA, OEAF and TOGAF are most preferable methodologies for developing any enterprise systems. The requirements for Trust Frameworks to build reference model can be accomplished through the usage of these three methodologies. However, ZF is one of the most adopted methods in the industry with its long history and fundamental perspectives for developing enterprise systems.

From the Figure 15, the graph explains the graphical strengths of each framework with numerical measurement on y-axis. The x-axis denotes frameworks with all the attributes and y-axis measures the total of each attribute in the corresponding framework. This shows the clear view of the most powerful or relevant framework for the reference model.

This research study focuses the most powerful method that offers highest non-functional requirements to address the architectural attributes for Trust Frameworks as mention in Chapter 4. The final decision goes more toward the TOGAF, as it is open for most the architectural attributes with highest ability to go "**agile**". This makes TOGAF to have ability to build the system that ensures required quality and

performance has been met at product or service level. But it's true in the sense that MDA and OEAF can be considered because of their latest version release.

Finally, the selected methodologies, TOGAF, MDA and OEAF depends on organizational goals, mission, culture and its architectural principles which. These models can be used in the development of reference model for Trust Frameworks after they under go agile development process. This evaluation shows that the TOGAF is the most closed to the requirements of the Trust Frameworks.

Chapter 9 Discussion: Adaptive enterprise service system

9.1 Introduction

This chapter is the second phase of this research to design reference model for Trust Frameworks. Our second phase requires solution to design agile enterprise system. Using the selected framework that can follow agile development is the core objective of this research. This chapter introduces the solution for agile development called Gill Framework, which helps enterprise architecture frameworks to develop agile systems. This is an addition step, solution to design reference model that is adaptive and governs all enterprise architectures. Dr. Asif Q. Gill has created a tool to design adaptive systems, from agile enterprise architectures frameworks.

Gill Framework is an extensive action-design research to explore agile, system thinking, service science and supply chain theories, frameworks, future technologies and best practices in industry in the context of enterprise strategy, architecture, service and project management [53]. This framework is applicable to ensure the agile activities in any adaptable enterprise systems. This open-source adaptive meta-framework has the capabilities for assessing, designing and transforming the agile enterprise as an adaptive enterprise service system.

Gill framework is a set of key integrated adoptive disciplines for developing and managing adaptive enterprise architectures for enabling enterprise-wide agility or adaptation with two main layers: Inner and outer layer.

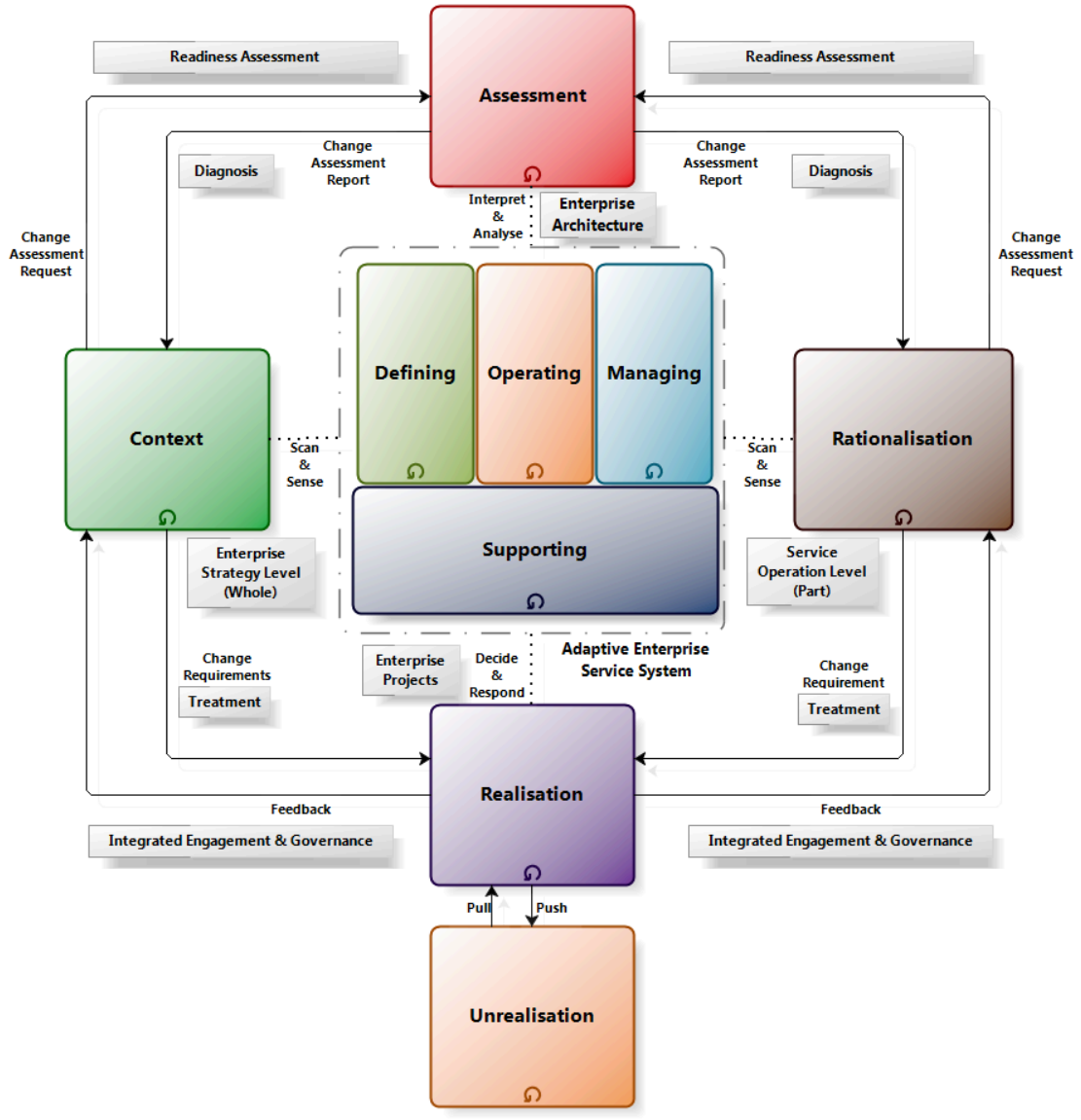

Figure 9.1: The Gill Framework [53][54].

Above figure 14.1 show the outer layer defines the adapting capability whereas the inner layer is the defining, operating, managing, and supporting capabilities. The inner layer defines an enterprise as a living system of service system known as "adaptive enterprise service system architecture". The combined model of these two layers is called "ADOMS" approach. The outer layer knows and passes the adaptation opportunities or changes to inner layer that manages the changes through integrated agile or adaptive capabilities [53][54]. ADOMS approach is made of following five capabilities:

- Adapting

- Defining
- Operating
- Managing
- Supporting

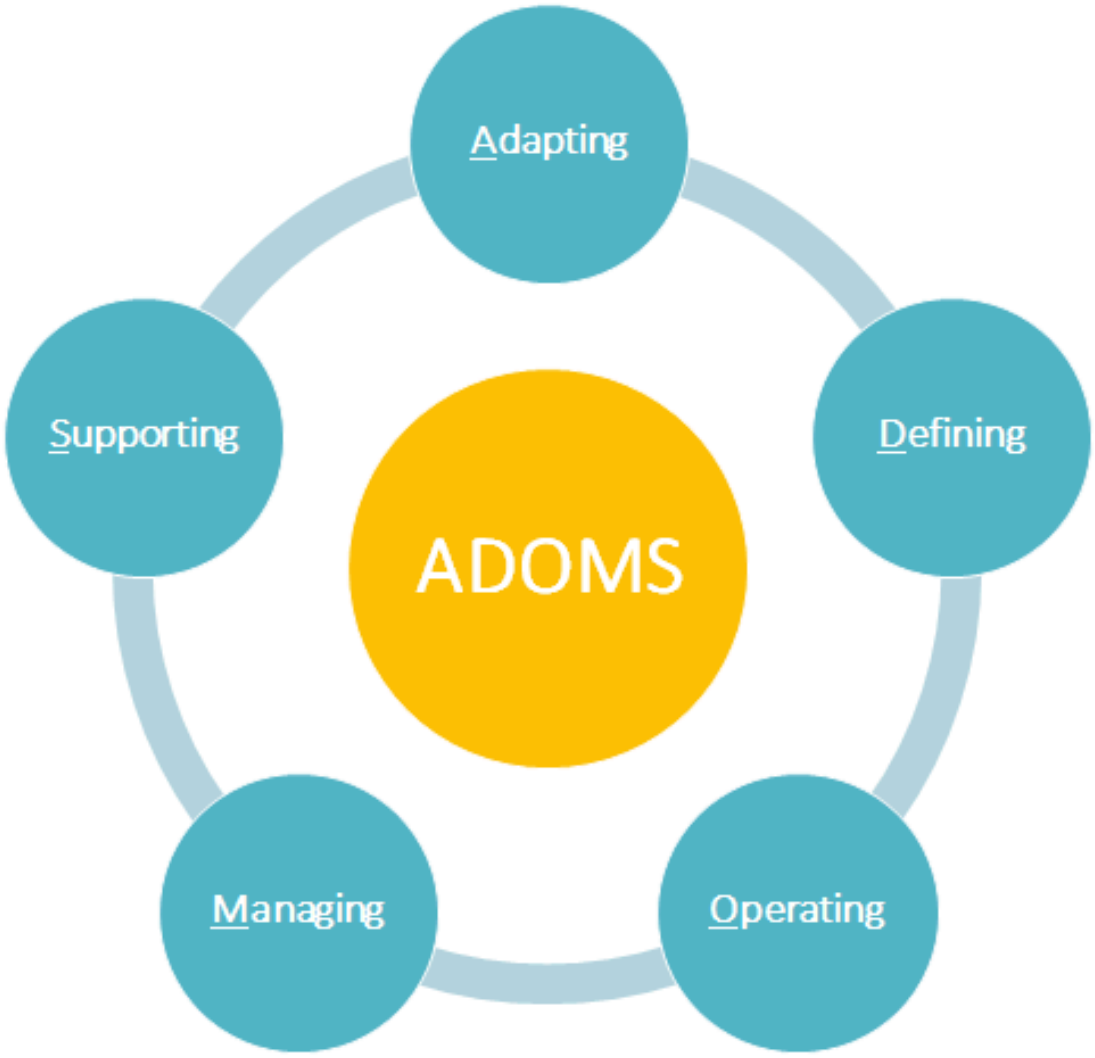

Figure 9.2: The Gill Framework – ADOMS [53][54].

Adapting – This outer layer capability offers services to scan, sense, interpret, analyse, decide and respond to internal and external changes. It offers service like context awareness, enterprise architecture assessment, rationalisation, realisation, and unrealisation. The outer layer is to find the change requirements that may initiate enterprise project(s), which are managed in the inner layer.

Defining – This is the integrated agile or adaptive capabilities for identified details such as adaptive enterprise strategy, architecture, project, service and requirements management capabilities.

Operating – This defines and integrates adaptive capabilities for value co-creation. It operates adaptive enterprise strategy, architecture, project, service and requirements management for services.

Managing – This manages the adaptive capabilities and their artefacts. It manages artefact changes through adapting.

Supporting – This is to provide support other capabilities like financial services.

9.2 Adaptive enterprise service system model

The adaptive enterprise service system (AESS) model outlooks the enterprise as an "Adaptive enterprise Service System". It may have one or multiple adaptive service systems with agility, services and multi-agent system as in figure 9.3

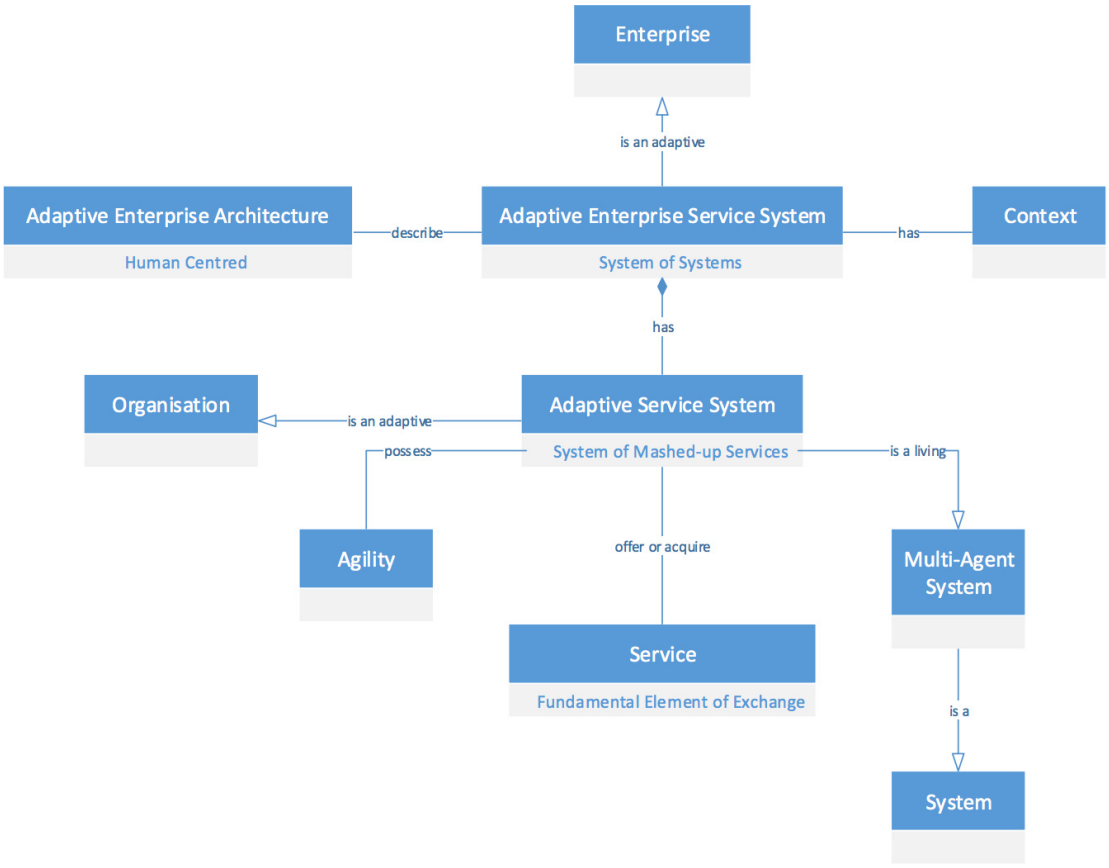

Figure 9.3: The AESS Conceptual Model [31]

A human cantered adaptive enterprise Architecture can explain an adaptive enterprise service system. It has context in term of principles, requirements, strategy, and policy with echo-system of one or many adaptive service systems. It is an adaptive organization as human, capability, function that possesses agility.

Agility of an adaptive service system can be verified by five agility concepts in Table 15. It explains that agile needs to be responsiveness, flexibility, speed, leanness and learning.

Systems thinking perspective for adaptive service system can be described in terms of eight key systems thinking concepts Table 15. The table verifies that it has to be system, autonomous, interdependent, integrated, context aware, adaptive, self-organizing, and lifecycle.

Service in adaptive service system can be described with eight key concepts in Table 15 [31]. Service should be offering, use, abstraction, interaction, access, governance, stakeholder and concern. A service is essential component in service system interactions.

Adaptive Enterprise Service System Model	Concepts	Enterprise Context	Enterprise Architecture Scope
Adaptive Service System			
Agility	Responsiveness Flexibility Speed Leanness Learning	Apply the agility concepts and understand the enterprise agility context for any EA work.	What are the agility requirements that are in scope for the purpose of an EA work? How would it impact the agility, other systems and services of the enterprise?
System	System Autonomous Interdependent Integrated Context Aware Adaptive Self-organizing Lifecycle	Apply the system concepts and understand the enterprise-as-a-whole integrated echo-system of systems context for any EA work.	What are the key systems that are in scope for the purpose of an EA work? How would it impact the agility, other systems and services of the enterprise?
Service	Offering Use Abstraction Interaction Access Governance Stakeholder Concern	Apply the service concepts and understand the enterprise-as-a-whole integrated echo-system of service systems context for any EA work.	What are the key services that are in scope for the purpose of an EA work? How would it impact the agility, other systems and services of the enterprise?

Table 15: Adaptive enterprise service system conceptual elements

The AESS conceptual model provides a complete approach to outline the enterprise context and the scope for adaptive enterprise. The components in AESS model ensure enterprise – it is a complete architecture work to design adaptive system.

So finally with the concept of system thinking, agility and service science, an adaptive enterprise service system [31] is:

"*A set of independent adaptive service systems that are integrated from independent systems which are geographically dispersed, dynamic contextual demand and have emergent behaviours.*"

Gill Framework here allows developing the adaptive enterprise system from the agile frameworks like TOGAF, MDA and OEAF. This chapter concludes with the AESS model to develop the enterprise systems to provide services with agile environment. This model enhances the concept of the adaptive enterprise system, which the core requirement for the reference model of Trust framework.

9.3 Result and Discussion

This section is to combine the series of research from top till the end. This is introduces a reference model for the Trust Framework. The purpose was to suggest a reference model for Trust Frameworks, which could be any of the enterprise architecture frameworks as they are used to design complex systems for enterprises.

This study uses different enterprise architecture attributes with their perspectives/aspects to analyse architecture framework. It shows that all architecture frameworks support the purpose of software architecture development. Most of them all adopt architecture planning, evolution and system interoperability. They have distinctive views for architecture modelling and have singular degrees in their views.
ZF lacks of detailed description of framework, which makes it difficult to further analyse. It shows that architecture models cannot be verified because of lack in Design Trade-offs, Design Rationale and Architecture. There is no any evidence to distinguish between architecture activities and its comprehensive design deeds. All listed enterprise architecture frameworks have little description of architecture design rationale even though they are essential aspect.

This research was occurred to select on of enterprise architecture framework, which is extended in depth study for agile nature. Result in chapter 9 shows that TOGAF can be used to as reference model due to its higher achievement in non-functional requirements measurement. Other options could be MDA and OEAF as they are similar in nature and provides similar features for system development. Except, they are limited in some section in some distinct structures and definitions. TOGAF offer open-end solution for most of the domains as it keeps them non-defined.

The major focus of TOGAF is Architecture Development Method (ADM), which has adapting nature and describes architecture scope and integration with development lifecycle. This process has been described in chapter 7 in details above. Together with this method, it has already agile nature to adapt the changes. This makes TOGAF first selection as a reference model to build Trust Frameworks. But some Trust Frameworks may need to be adaptive in nature rather than only agile. This introduces another development process of system called AESS Model from Gill Framework of Dr. Gill.

Now, Finally together TOGAF and AESS model gives the complete model for reference model to build Trust Frameworks. This combine approach of TOGAF to guide the enterprise architecture and AESS model to develop system with an adaptive nature concludes research work to build reference model.

This reference model could be the combine of MDA or OEAF with AESS model to design adaptive enterprise system. This depends upon the domain of Trust Frameworks. Evaluation in chapter 8 has already mentioned that, none of framework is perfectly ready for new system development. They all have to be reference in some sense to develop them effectively, quickly and less costly.

Now, depending upon the business requirements, objectives and domain of Trust framework, either of TOGAF, MDA and OEAF can be combined with AESS model to create reference model for developing Trust Frameworks.

Chapter 10 Conclusion

The research suggests a reference model to build Trust Frameworks for Identity system purpose in "*National Strategy for Trusted Identities in Cyberspace*". The selection process was started with a comparative analysis of enterprise architecture frameworks. The comparative analysis was done based on the measurement attributes and scale mention in Chapter 5 to verify attributes of adaptive service system from Chapter 4. Finally selected frameworks can be tailored with AESS model to create a reference model for Trust Framework. This study includes fourteen enterprise architecture frameworks to be verified their architecture attributes and non-functional requirements.

The purpose of the study is to answer research questions generated in Chapter 3, which have asked to suggest a reference model for the Trust Frameworks.

- Which existing enterprise architecture framework could be the possible reference model based on the attributes?
- What are agility, interoperability, service science and systems thinking concepts for enterprise architecture frameworks?
- What could be the possible solution to develop adaptive service systems from agile enterprise architecture frameworks?
- What could be the possible solution to develop Trust Frameworks in Identity Ecosystem proposed by "National Strategy for Trusted Identities in Cyberspace"?

To conclude these questions, the result and discussion in chapter 9 finally gives the output as possible reference model. There could be many Trust Frameworks used in the Identity Ecosystem, which may need various frameworks to develop. But if there is reference model to reference, it will provide the guidelines for every architectural development, methods to be use, process to be implementing, inputs to be taken, expected outputs and address proper business requirements. This concludes research with suggestion of combined model of TOGAF and AESS method for developing adaptive service system. This can be improved with development of distinctive new

model for future research purpose. The extension of this research is the possibility of this reference model in the real projects. The combination of TOGAF along with AESS model to design the system for integrated enterprise systems of government and agencies is the best practical approach. The research could be the more in depth to declare the attributes and find the reference model to required domain.

References

[1] Jeanne W. Ross, Peter Weill, and David C. Robertson (2006) Enterprise Architecture As Strategy: Creating a Foundation for Business Execution. Harvard Business Review Press.

[3] MIT Center for Information Systems Research (2014), Enterprise Architecture (http://cisr.mit.edu/research/research-overview/classic-topics/enterprise-architecture/), Accessed Sept 04, 2014.

[4] Oracle (2014), Oracle White Paper in Enterprise Architecture -The Oracle Enterprise Architecture Framework. (http://www.oracle.com/technetwork/topics/entarch/oea-info-arch-framework-dev-process-513866.pdf). Accessed. Sept 03. 2014

[5] Clay Richardson, Forrester Blogs – Design Thinking Reshapes EA For Dynamic Business, (2013) (http:/ / blogs. forrester. com/ clay_richardson/ 13-04-12-design_thinking_reshapes_ea_for_dynamic_business). Accessed Sept 02, 2014

[6] Joe McKendrick, ZDNet – Gartner urges more 'design thinking' to break enterprise architecture out of its silo, (2010) (http:/ / www. zdnet. com/ blog/ service-oriented/ gartner-urges-more-design-thinking-to-break-enterprise-architecture-out-of-its-silo/ 4858). Accessed Sept 02, 2014

[7] Mentz, J, Van der Merwe, Alta, & Kotze, Paula. (2012) " A Comparison of Practitioner and Researcher Definitions of Enterprise Architecture using an Interpretation Method (http:/ / hufee. meraka. org. za/ Hufeesite/ staff/ the-hufee-group/ paula-kotze-1/ publications/ confenis2011_submission_74. pdf)". Accessed Sept 05, 2014

[9] Tang, A., Han, J., Chen, P. (2004). A Comparative Analysis of Architecture Frameworks, School of Information Technology, Centre for Component Software & Enterprise Systems, Swinburne University of Technology

[10] Zachman, J. (1999), A Framework for Information Systems Architecture. IBM Systems Journal, 38 (2-3), 454-70

[11] Federation of EA Professional Organizations, Common Perspectives on Enterprise Architecture (http:/ / feapo. org/ wp-content/ uploads/2013/ 11/ Common-Perspectives-on-Enterprise-Architecture-v15. pdf), Architecture and Governance Magazine, Issue 9-4, November 2013. Accessed Sept 05, 2014

[12] John Zachman (2008), John Zachman's Concise Definition of The Zachman Framework™, Zachman International, Inc.

[13] Sowa, J. & Zachman, J. (1992). Extending and Formalizing the Framework for Information Systems Architecture, IBM Systems Journal, 31(3), 590-616

[14] Durward P. Jackson (1992), "Process-Based Planning in Information Resource Management". In: Emerging Information Technologies for Competitive Advantage and Economic Development. Proceedings of 1992 Information Resources Management Association International Conference. Mehdi Khosrowpour (ed). ISBN 1-878289-17-9

[15] The Open Group (2011) TOGAF Version 9.1 http://pubs.opengroup.org/architecture/togaf9-doc/arch/

[16] DoD, (2011). DoD Architecture Framework Version 2.0 (http://dodcio.defense.gov/TodayinCIO/DoDArchitectureFramework.aspx). Accessed Sept 06, 2014

[17] Federal Enterprise Architecture (FEA), (2012). The common approach to federal enterprise architecture,

(http://www.whitehouse.gov/sites/default/files/omb/assets/egov_docs/common_approach_to_federal_ea.pdf) Accessed on 09 Sept, 2014

[18] Federal Enterprise Architecture (FEA) (2013), Federal Enterprise Architecture Framework Version 2, (http://www.whitehouse.gov/sites/default/files/omb/assets/egov_docs/fea_v2.pdf) Accessed on 10 Sept, 2014

[19] Sherwood J., Clark A., and Lynas D.,(2009). Enterprise security architecture whitepaper. SABSA Limited, 2009.

[20] Cabreraa D., L. Colosic, and C. Lobdellc, (2008). Systems thinking, Evaluation and Program Planning, Volume 31, Issue 3, August 2008, Pages 299-310, ISSN 0149-7189, http://dx.doi.org/10.1016/j.evalprogplan.2007.12.001.

[21] Sherehiy B., Karwowski W., Layer J. K.,(2007). A review of enterprise agility: Concepts, frameworks, and attributes, International Journal of Industrial Ergonomics, Volume 37, Issue 5, May 2007, Pages 445-460, ISSN 0169-8141, http://dx.doi.org/10.1016/j.ergon.2007.01.007.

[22] Vernadat F.B.,(2007). Interoperable enterprise systems: Principles, concepts, and methods, Annual Reviews in Control, Volume 31, Issue 1, 2007, Pages 137-145, ISSN 1367-5788, http://dx.doi.org/10.1016/j.arcontrol.2007.03.004.

[23] Shariati M., Bahmani F., Shams F., (2011). Enterprise information security, a review of architectures and frameworks from interoperability perspective, Procedia Computer Science, Volume 3, 2011, Pages 537-543, ISSN 1877-0509, http://dx.doi.org/10.1016/j.procs.2010.12.089.

[24] Kruchten, P.B.,(1995).The 4+1 View Model of architecture, Software, IEEE , vol.12, no.6, pp.42,50, Nov 1995, doi: 10.1109/52.469759

[25] Tang, A., Han, J., Chen, P., (2004). A comparative analysis of architecture frameworks, Software Engineering Conference, 2004. 11th Asia-Pacific , vol., no., pp.640,647, 30 Nov.-3 Dec. 2004

[26] Winter, R., Fischer R., (2006). Essential Layers, Artifacts, and Dependencies of Enterprise Architecture," Enterprise Distributed Object Computing Conference Workshops, 2006. EDOCW '06. 10th IEEE International , vol., no., pp.30,30, 16-20 Oct. 2006

[27] Wilkinson, M., (2006). Designing an 'adaptive' enterprise architecture, Kluwer Academic Publishers-Consultants Bureau, 2006-10-01

[28] Minoli, D., (2008). Enterprise Architecture A to Z: Frameworks, Business Process Modeling, SOA, and Infrastructure Technology, Auerbach Publications, 2008

[29] Schekkerman, J., (2004). Extended Enterprise Architecture Framework (E2AF) Essential Guide, IFEAD, 2004

[30] Bittler, R. S., Kreizman, G., (2005). Gartner Enterprise Architecture Process: Evolution 2005, Gartner, 2005

[31] Gill, A. Q. (2014). Applying Agility and Living Service Systems Thinking to Enterprise Architecture. International Journal of Intelligent Information Technologies (IJIIT), 10(1), 1-15.

[32] MODAF, (2012). MOD Architecture Framework, Ministry of Defence, 12 December 2012

[33] Vallecillo, A.(2001). RM-ODP: The iso reference model for open distributed processing, dintel edition on software engineering, pp. 69–99 March 2001

[34] Leist, S., and Zellner, G., (2006). Evaluation of current architecture frameworks. In Proceedings of the 2006 ACM symposium on Applied computing (SAC '06).

ACM, New York, NY, USA, 1546-1553. DOI=10.1145/1141277.1141635 http://doi.acm.org/10.1145/1141277.1141635]

[35] Richmond, B. (1994), Systems thinking/system dynamics: Let's just get on with it. Syst. Dyn. Rev., 10: 135–157. doi: 10.1002/sdr.4260100204

[36] The white House, (2011). National Strategy for trusted identities in cyberspace: Enhancing Online Choice, Efficiency, Security, and Privacy,
The White House, Washington, April 2011.

[37] Sessions, R., (2007). A Comparison of the Top Four Enterprise-Architecture Methodologies, ObjectWatch, Inc, 2007.

[38] Odongo, A.O., Kang, S., Ko, I., (2010). A Scheme for Systematically Selecting an Enterprise Architecture Framework, Computer and Information Science (ICIS), 2010 IEEE/ACIS 9th International Conference on , vol., no., pp.665,670, 18-20 Aug. 2010, doi: 10.1109/ICIS.2010.85

[39] Schekkerman, J., (2011). Enterprise Architecture Tool Selection Guide, IFEAD, 2011

[40] Ritchot, B. (2013). An Enterprise Security Program And Architecture To Support Business Drivers. Technology Innovation Management Review, 3(8): 25-33. http://timreview.ca/article/713

[41] GERAM, (1999). GERAM: Generalised Enterprise Reference Architecture and Methodology, IFIP-IFAC Task Force, 1999

[42] Object Management Group, (2010). The MDA Foundation Model - The Architecture of Choice for a Changing World, Object Management Group, 2010

[43] Alexopoulou, N., Kanellis, P., Nikolaidou, M., & Martakos, D. (2009). A Holistic Approach for Enterprise Agility. In J. Gupta, S. Sharma, & M. Rashid (Eds.)

Handbook of Research on Enterprise Systems (pp. 1-18). Hershey, PA: Information Science Reference. doi:10.4018/978-1-59904-859-8.ch001

[44] Robin G.Q., (2009). Computational Thinking of Service Systems: Dynamics and Adaptiveness Modeling. Service Science 1(1):42-55. http://dx.doi.org/10.1287/serv.1.1.42

[45] Demirkan, H., Spohrer, J.C., Krishna, V., (2011). Introduction of the Science of Service Systems in The Science of Service Systems, Springer, pp 1-11, 2011, 10.1007/978-1-4419-8270-4_1

[46] Katzan, H.,(2011), Review of the Cyberspace Policy and Trusted Identity Documents, Review of Business Information Systems - Second Quarter 2011, Volume 15, Number 2, Accessed on 15 December 2015, http://www.cluteinstitute.com/ojs/index.php/RBIS/article/viewFile/4203/4268

[47] Gill, A. Q. (2013). Towards the development of an adaptive enterprise service system model.

[48] Lim, N., Lee, T. G., & Park, S. G. (2009, May). A comparative analysis of enterprise architecture frameworks based on EA quality attributes. In Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing, 2009. SNPD'09. 10th ACIS International Conference on (pp. 283-288). IEEE.

[49] Mohamed, M. A., Galal-Edeen, G. H., Hassan, H. A., & Hasanien, E. E. (2012, November). An evaluation of enterprise architecture frameworks for e-government. In Computer Engineering & Systems (ICCES), 2012 Seventh International Conference on (pp. 255-260). IEEE.

[50] Urbaczewski, L., & Mrdalj, S. (2006). A comparison of enterprise architecture frameworks. Issues in Information Systems, 7(2), 18-23.

[51] Odongo, A. O., Kang, S., & Ko, I. Y. (2010, August). A scheme for systematically selecting an enterprise architecture framework. In Computer and Information Science (ICIS), 2010 IEEE/ACIS 9th International Conference on (pp. 665-670). IEEE.

[52] Sessions, R. (2007). Comparison of the top four enterprise architecture methodologies.

[53] Gill A., Q., (2012), The Gill Framework - Adaptive Enterprise Architecture Toolkit, CreateSpace Independent Publishing Platform, 2012.

[54] Gill A., Q., (2015), The Gill Framework®, 2015, accessed from http://www.aqgill.com/

[55] Kasunic, M. (2001). Measuring systems interoperability: Challenges and opportunities. Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst. 2001.

[56] Folmer, E., & Verhoosel, J. (2011). State of the art on semantic IS standardization, interoperability & quality. UT, CTIT, TNO en NOiV. ISBN 9789090260303

[57] EIF. (2004). European interoperability framework, white paper. Brussels: February 2004. http://www.comptia.org.

[58] Van der Veer, H., & Wiles, A. (2006). Achieving Technical Interoperability - the ETSI approach. Sophia Antipolis Cedex: ETSI

[59] Dube M. R., and Dixit, S. K., (2011). Comprehensive measurement framework for enterprise architectures, IJCSIT, Vol 3, No 4, August 2011

[60] Thurmond, V. (2001). The point of triangulation. Journal of Nursing Scholarship, Vol. 33, Issue 3, pp. 254–256.

[61] Qumer, A. and Henderson-Sellers, B. (2008). An Evaluation of the Degree of Agility in Six Agile Methods and its Applicability for Method Engineering. Journal of Information and Software Technology (IST), Vol. 50, Issue 4, pp. 280-295.

[62] Miller, J.G. (1995). Living systems. University Press of Colorado.

[63] Spohrer, J. & Kwan, S. K. (2009). Service Science, Management, Engineering, and Design (SSMED): An Emerging Discipline - Outline & References. International Journal of Information Systems in the Service Sector, Vol. 1 Issue (3).