A Scheme for Systematically Selecting an Enterprise Architecture Framework

Agnes Owuato Odongo, Sungwon Kang and In-Young Ko
Kenya Electricity Generating Company (KENGEN),
Kenya

1. Introduction

EAF is "a logical structure for classifying and organizing the descriptive representations of an enterprise that are significant to the management of the enterprise as well as to the development of the enterprise's systems" [17]. EAF is also defined as "a set of assumptions, concepts, values, and practices that constitute a way of viewing reality" [27]. It establishes data element definitions, rules, and relationships and a baseline set of products for consistent development of systems, integrated, or federated architectures. These architecture descriptions may include Families of Systems (FoSs), Systems of Systems (SoSs), and net-centric capabilities for interoperating and interacting in the Net-Centric Environment" [6]. Various EAFs have been developed such as Zachman Framework (ZF) [13][14], Department of Defense Architecture Framework (DoDAF) [6], The Open Group Architecture Framework (TOGAF) [36][35], Federal Enterprise Architecture Framework (FEAF) [10][11], Treasury Enterprise Architecture Framework (TEAF) [8], and The Command, Control, Communications, Intelligence, Surveillance, and Reconnaissance (C4ISR) [29]. However, their unique limitations and dissimilarities make it impossible to use any of them in isolation to completely solve a specific problem. In short, no single existing EAF in isolation is capable of providing a complete solution to an Enterprise Architecture system design problem [31][12][19] [3].

Comparisons of EAFs have been done in the past in order to select the best EAF for system or Enterprise Architecture (EA) development [3][32][31][27][28][24][22][26]. However, none of the past comparisons have focused on EAFs usages and related perspectives and aspects in comparing and selecting EAF based on how they support the various usages. A complete EA system design problem solution may encompass various usages and choosing the right EAF that best supports each usage is of paramount importance. Research has revealed that the existing EAFs have weaknesses and strengths. This is to say that none of them in isolation can offer complete EA design support solution [2][19][17]. EAFs may overlap by addressing similar views, however, they are developed for particular needs, and they differ in the stakeholders they support and their domain. In regard to the above statement, EAFs is used in EA design in order to manage system complexity and to align IT to business [27]. The organizations not using EAFs in EA design spend more in building IT systems which are costly to maintain and offer little business value. It is a fact to say that failing to use EAFs in EA design that aligns IT with business needs makes it expensive and difficult to organize and align IT systems with business needs later after realizing the alignment was not met initially. Comparing and selecting the best EAF that best supports a specific usage is meant to reduce complexity in system management and to allow alignment of IT systems to business needs at initial stage.

This chapter is unique in that it has identified the various usages and considered them to select the best EAF for each. Usages have not been used in selecting EAF In comparing and selecting EAF in past, therefore, this is a new step to improve the selection of EAF for use. All existing EAFs in isolation cannot offer complete EA design solution to an enterprise problem [27] and neither is any of the existing EAF in isolation can support all the listed usages in our chapter as has been demonstrated by the two examples given in section 5. The chapter, therefore, makes a step further by proposing an EAF selection scheme and uses the scheme in selecting an EAF for each usage based on how it addresses the perspectives and aspects relevant to the usage. By selecting the relevant usages to the specific situation and selecting the best EAFs supporting them provides the best solution to the EA systems design. This new scheme has not been availed by the researchers in the past.

To select the best EAF for system or EA development, EAFs have been compared in the past [2][11][24]32]. Past comparisons did not focus on EAF usages and related them to perspectives and aspects. Perspectives are comparison criteria and aspects are the criteria attributes. The chapter proposes an Enterprise Architecture Framework selection scheme (EAFSS) focusing on usages and uses it to select the best EAF for each usage. The scheme has four steps as shown in Fig. Step 1 is split into Fig.1. step 1 is split into 1A and 1B. 1A focuses on identifying usages and 1B searches for comparison perspectives and aspects.

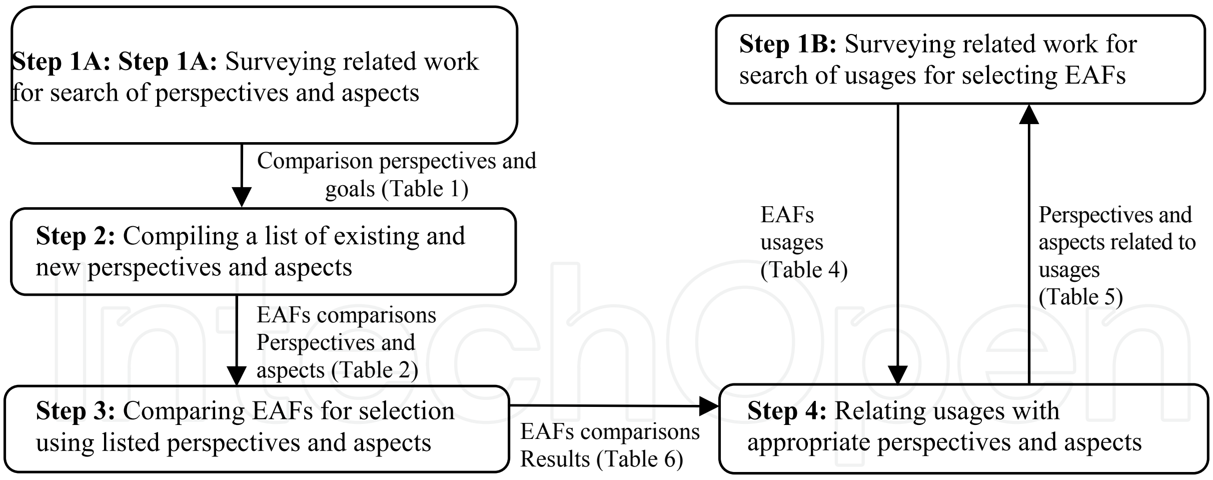
Step 1A: Step 1A: Surveying related work for search of perspectives and aspects
Comparison perspectives and goals (Table 1)
Step 2: Compiling a list of existing and new perspectives and aspects
EAFs comparisons Perspectives and aspects (Table 2)
Step 3: Comparing EAFs for selection using listed perspectives and aspects
EAFs comparisons Results (Table 6)
Step 1B: Surveying related work for search of usages for selecting EAFs
EAFs usages (Table 4)
Perspectives and aspects related to usages (Table 5)
Step 4: Relating usages with appropriate perspectives and aspects

Fig. 1. The chapter approach steps

The focus of the steps is as explained. Step 1A focuses on surveying related work in search of perspectives and aspects. Step 1B conducts the survey on related work with the aim of determining EAF usages. Step 2 compiles the list of perspectives and aspects identified in Step 1A. Step 3 compares the EAF using the listed perspectives and aspects. Step 4 relates the usages to the perspectives and aspects. All these steps have their results in the form of tables to be shown later in this chapter.

The remainder of the chapter is structured as follows: Section 2 describes related works on EAFs comparisons and other EAFs related work. Section 3 focuses on steps toward comparing EAFs. Section 4 introduces our proposed Enterprise Architecture Framework Selection Scheme (EAFSS). Section 5 is the comparisons of EAFs. Section 6 gives examples of application of EAFSS. Section 7 concludes the research and outlines our contributions towards EAFs selection. Section 8 gives the references used in the chapter.

2. Related works

This section surveys related works. It is divided into two parts. The first part covers related works on comparison of EAFs. The second part covers other related works which we thought would contribute positively towards the chapter's goal.

2.1 Related Works on EAF comparison

Comparison was conducted on EAF to develop a method of selecting EAF for system or EA development in specific environments [3]. Furthermore, EAFs are used in designing EA to see the whole enterprise as well, in order to identify the commonality as well as recognize the dimension of entirety and of depth [16]. The comparison focused only on characterizing EAFs by vital elements they need to contain to satisfy the system or EA development. EAFs were compared using ZF as a benchmark, views, abstractions and system development life cycle. The goal was to provide a means of selecting the best fitting EAFs [22]. To determine the current state of EAFs, [31] compared the EAFs using the same perspectives as [3] but considered very few aspects under the three perspectives. He focused his comparison on selecting or tailoring or adapting an EAF [31]. Comparison on EAFs was conducted to determine the contributions each EAF has made to improve EA efforts [26]. The results show that EAFs contribution varies.

Architecture Framework Ontology (AFO) was used to compare EAFs with the goal of characterizing them to determine their overlaps [24]. The goal was to identify certain features that separate artifacts from products they produce to be able to asses EAF for suitability for system or EA development. Comparison was conducted on EAFs by recognizing features of the EA discipline and their principles as relevant; by focusing on structures and methodologies [27]. This was to enable a blending method of developing EAF or selecting the best for specific use. Comparison was conducted on EAFs using requirements that EAF should meet to develop, describe and maintain EA, and determine EAFs capabilities in supporting EA [32]. Commonly used EAFs were compared using some quality aspects to identify relationship among views and the usage growth of EAFs [28].

2.2 Other critical related works

A survey was conducted on papers touching on EAF and the result was that adoption is the major topic [19]. Other topics included benefits of using EAFs, common modeling techniques, summarizing and using EAFs and enterprise environmental change design principles. Assessment of EAFs was conducted to develop an extended EAF by identifying EAFs products to form extended EAF [12]. The topics intended to improve EA effort and is eagerly awaited for. Companies in diverse industrial domains own very complex Enterprise Software Systems (ESS) that has become a challenge to manage and so a utility-based approach was proposed to replace Chief Information Officer (CIO) to manage complexity in ESS [23].

Survey on EA trends revealed that EA is used worldwide in industries, minor and sizeable organizations, and in government organizations [15]. The stormy subjects include delivering road maps for change; managing IT portfolio and complexity; and supporting system development. Modeling EA using BPML standard business methods is widely acknowledged, however, the current accepted Information Systems (IS) modeling standards are OMG's Model Driven Architecture and UML OMG's [18]. An assessment was conducted on EAFs to determine the status of EAFs in regard to industry and government IT; to identify an EAF definition that can be commonly used as a reference [25]. Architecture-centric concept was provided to model the enterprise to address complete concerns.

2.3 Summary of the related work

Related work on comparisons covered various perspectives and aspects. A perspective focuses on a specific interest on an object. It is used to classify content of an object by categorizing them into groups that have related aspects. Aspects are the contents connected to the perspective. Comparison goals are Are the outcomes of research work. Table 1 is a summary of the related works on EAFs comparison. In the last column we have perspectives and aspects, for example, [3] used three perspectives for comparison, i.e. P1, P2 and P3 and the P1 perspective has the aspects P1.1, P1.2 and etc. In the table, when the same perspective or aspect is repeated in the following papers the same ID for it is repeatedly used.

Paper	Comparison Goal	EAFs Compared	Comparison perspectives and aspects
[3]	Selecting and tailoring EAF for system or EA development.	RM-ODP 4+1 View TOGAF FEAF ZF	P1. Goals P1.1 Architecture definition and understanding P1.2 Architecture development process P1.3 Architecture evolution support P1.4 Architecture analysis P1.5 Architecture models P1.6 Design rationale P1.7 Design tradeoff P1.8 Standardization P1.9 Architecture knowledge base P1.10 Architecture verifiability P2. Inputs P2.1 Business drivers P2.2 Technology inputs P2.3 Business requirements P2.4 Information system environments P3.5 Current architecture P2.6 Non functional requirements

Paper	Comparison Goal	EAFs Compared	Comparison perspectives and aspects
			P3. Outcomes P3.1 Business model P3.2 System model P3.3 Information model P3.4 Computation model P3.5Software configuration model P3.6 Software processing model P3.7 Implementation model P3.8 Platforms P3.9 Non functional requirements design P3.10 Transition design P3.11 Design rationale
[22]	Selecting and determining best fit of EAFs.	DoDAF ZF TEAF FEAF TOGAF	**P4. Views** P4.1 Planner P4.2 Owner P4.3 Designer P4.4 Builder P4.5 Subcontractor P4.6 User **P5. Abstractions** P5.1 What P5.2 How P5.3 Where P5.4 Who P5.5 When P5.6 Why **P6. System development life cycle** P6.1 Planning P6.2 Analysis P6.3 Design P6.4 Implementation P6.5 Maintenance
[32]	Identifying requirements to be met by EAFs for EA development, determining EAF capability support to EA building, and developing EA descriptions.	ZF ARIS C4ISR DoDAF FEAF MDA TEAF TOGAF	**P7. Guide** P7.1 Meta model P7.2 Procedure model P7.3 Modeling technique P7.4 Role P7.5 Specification document

Paper	Comparison Goal	EAFs Compared	Comparison perspectives and aspects
[27]	Developing EAFs by blending method, as well as selecting an EAF for use	ZF FEAF GEAF TOGAF	**P7. Guide** P7.6 Process completeness P7.7 Maturity model P7.8 Reference model guidance P7.9 Practice guidance P7.10 Governance guidance P7.11 Partitioning guidance P7.12 Prescription catalog **P8. Quality** P8.1 Taxonomy completeness P8.2 Vendor neutrality P8.3Time to value P8.4 Information availability P8.5 Business focus
[28]	Evaluating relationship among the views and the EAFs usage growth.	DoDAF FEAF TOGAF	**P8. Quality** P8.6 Alignment P8.7 Integration P8.8 Value creation P8.9 Change management P8.10 Compliance
[26]	Identifying appropriate EAF and incorporating best practices to enhance EA effort.	FEAF TOGAF TEAF DoDAF	**P7. Guide** P7.6 Guidance on EA development process P7.7 Maturity/status **P9. Miscellaneous** P9.1 Basic organizational approach and views P9.2 Integrated EA product specifications descriptions P9.3 EA strategic vision and goals P9.4 Architecture principles provision P9.5 Product for specifying standards P9.6 Security considerations in EA P9.7 Tool support P9.8 EA repository issues
[24]	Identifying EAFs features for EA suitability, selecting the best framework and identifying similarities in EAFs	FEAF DoDAF TEAF TOGAF ZF GERAM	**P1. Goals** P1.3 Architecture evolution support **P3. Outcomes** P3.10 Transition strategy and plan definition **P8. Quality** P8.11 Explicit detail of products **P9. Miscellaneous** P9.9 Explicit specification

Paper	Comparison Goal	EAFs Compared	Comparison perspectives and aspects
			P9.10 Architecture domain P9.11 Analytical approach P9.12 Stakeholder P9.13 Application domain development process
[31]	Tailoring, selecting and adapting EAFs.	ZF TOGAF C4ISR	**P1. Goals** P1.1 Architecture definition and understanding P1.2 Architecture process P1.3 Architecture evolution support P1.8 Standardization P1.9 Architecture knowledge base **P2. Inputs** P2.1 Business drivers P2.2 Technology support **P3. Outcomes** P3.1 Business model P3.10 Transitional design **P9. Miscellaneous** P9.7 Visualization tool P9.14 Conformance

Table 1. A summary of comparison perspectives and goals

2.4 The need for further research on selecting EAF for specific usage

Regardless of the research done by [7] [32] [31][27], further research like the one covered by this chapter is still needed because the systems are becoming more complex and delivering less business value. Focusing on developing better methods of assessing and selecting appropriate EAF to develop EA is critical in enhancing enterprise complexity management, and increasing chances of delivering systems which have more value to the business [27]. Complex systems and EA development requires more specific planning that calls for more comprehensive comparison assessment of the EAFs based on usages we identified, and this will support the comparison and selection of EAF. Most of existing comparisons are limited in scope and usages have not been considered as a basis for EAF selection. For example, only objectives, inputs and outputs perspectives were used [3], [Session 07] selected some criteria for comparison EAF fitness in different applications, [24] used ontology to characterize EAFs, [26] considered EAF contributions, [28] considered only three layers of EAFs, [31] used the same perspective as [27] but with fewer aspects for each, [32] considered contributions each EAF has made towards architecture development support, and [22] used views, abstractions and system development life cycle to compare EAFs. None of these authors considered usages and how EAF support them.

Usages may need to be combined to get the complete architecture design solution and hence the EAFs supporting these usages should be used to get a valuable solution in relation to EA design. Researchers that compared the EAFs in the past did not use weighted comparison criteria except [27]. It is difficult to use non-weighted criteria to compare the EAFs for selection. Although this author used the weighted comparison criteria and talked of blended methodology, he did not provide a step by step process for selecting the best EAF.

Past research reveals that authors used different methods in comparing the EAFs. For example, context of a fictional company having functional operational problems was used as an example of blended methodology by integrating best practices of EAFs [27], [3] used goals inputs and outputs to compare EAFs, [26] used contributions that EAF have made to improve EA effort, [24] used ontology to characterize EAFs, [31] focused on the current state of EAFs, [22] focused on views and aspects in comparing EAFs, [28] focused on EAFs strategic support to IT challenges and decisions, [32] focused on requirements that must be provided by EAFs. However, the assessments conducted in this chapter are special by making use of selected formats that are subjected to all the selected EAFs in order to verify their strengths and weaknesses. The formats used in our assessment include philosophy, dimension of EAF, structure, artifacts, enterprise information system development process, and EAF strengths and weaknesses. We believe that our selection scheme is better compared to the past ones.

This chapter proposes a selection scheme and considers all the perspectives used in the previous comparisons as well as added new perspectives to enhance EAF assessment for more improved and balanced comparisons. A suggestion is given for use of comparison in Table 7, and allowance is given for expanding the table by additional of perspective that can be compared by the user of the chapter or shrinking the table by removing what is not relevant to the user. The scheme we developed has the ability to significantly assess the EAFs and subsequently select the best EAF for each usage we identified in this chapter. The scheme is intended to significantly improve EAF selection. The scheme is used to make use of existing EAFs to avoid developing a new one and to save time and money. Examples are given using some selected usages and perspectives to demonstrate the application. The two cases demonstrated prove that the chapter can be generally used for assessing and selecting EAF for any enterprise.

3. EAFs comparison process steps

There is need to develop tables which are referenced when applying the EAFSS selection scheme in Figure 1. Therefore, this section develops the needed tables for EAFs selection purposes. We identify EAFs to compare, conduct EAFs survey, identify perspective and aspects, determine scaling scheme and finally conduct EAFs comparisons. Below are our steps in achieving the goal. This section identifies EAFs to be compared using some criteria, EAFs assessment criteria and of compares EAFs.

3.1 Selecting EAFs for comparison

It is a fact to say from the literature that ZF, DoDAF, TOGAF, FEAF, and TEAF are the most dominant EAFs and have been used the longest [3]. FEAF is the most complete methodology with ZF-like classification and a TOGAF-like structural design process [2]. FEAF, TEAF, and DoDAF are the common federally sponsored and accepted EAFs [2].

C4ISR was dropped and replaced by DoDAF, GERAM was dropped and replaced by TOGAF because TOGAF is a general EAF and used in any enterprise with modifications [36]. Open Distributed Processing – Reference Model (RM-ODP) was dropped because its commonality with other EAFs is limited.

3.2 Conducting survey on selected EAFs

An enterprise architecture developed for use in developing an enterprise system is the structure or structures of that system, which comprise system elements, the externally visible properties of those elements, and the relationships among them [21]. In this regard, it is necessary to identify the formats that can be used to compare the EAFs in relation to this definition so that we can understand in detail the differences, similarities and the limitations of these EAFs.

3.2.1 Criteria used in conducting survey on EAFs

To present summaries of the major EAFs to be compared, we summarized the following six presentation format items, 1) philosophy, 2) dimension of EAF, 3) structure, 4) artifacts, 5) enterprise Information System (EIS) development process, and 6) EAF strengths and weaknesses.

Details of Presentation Formats

The survey is conducted to understand the relationships, commonalities, dissimilarities, and incompleteness among EAFs, [22]. This will improve EAF selection accuracy, assists to rate EAFs by assigning weights to aspects. This is aimed at saving time and money in developing a new one from scratch. The following six formats were used to evaluate the selected EAFs:

F1) Philosophy: Focuses on the purpose of developing EAFs and goals expected to be achieved; and this may include designing Enterprise System Architecture (ESA), supporting realization of goals, identifying enterprise business potential investment areas, and developing enterprise IT system.

F2) Dimensions of the framework: These are classifications, definitions and presentations of architectural artifacts.

F3) Structure of the framework: This holds basic EAFs items and information about them. It provides theories and outlines performed in business processes and how they are performed.

F4) Artifacts: Final products that describe the enterprise functionalities, interactions and requirements.

F5) Architecture development process: It is a series of steps employed in creating a structural design for an enterprise. The process is followed in planning, analyzing, designing, developing and maintaining the created architecture.

F6) Strengths and weaknesses: EAFs have strengths and weaknesses. The weaknesses limit EAFs usage for achieving certain goals. Strengths enhance its usefulness in achieving expected goals. For summaries of the major EAFs, see [1].

3.2.2 Assessments of EAFs

3.2.2.1 Zachman Framework (ZF)

F1) Philosophy: ZF presents architectural artifacts and focuses on business value and suppleness [13]. It also addresses requirement traceability, builds useful and understandable functionalities, and defines communication among stakeholders [31] [5].

F2) Dimensions of the framework: ZF is divided into two dimensional organizations [13].

The rows are roles for stakeholders, and the columns depict divisions of requests.

F3) Structure of the framework: ZF structure is built on 6 fundamental interactive interrogatives and views as shown in Table 3. The models relate to player groups giving a view of the business being modeled [13]. It has a total of 36 intersecting cells each meeting at a point between a player's perspective and a descriptive focus. A cell is a result of a stakeholder's performed architecture activity on a single system viewpoint.

F4) Artifacts: ZF has 30 outcomes from the 36 intersecting cells, with six views that describe cell items.

F5) Architecture development process: No direct guidance on the step-by-step process in developing the architecture, but the process uses viewpoint of the stakeholders [5]. The process consists of: 1) the individual concerned with the business in an organization 2) business management group 3) the business individuals experts to analyze the business systems 4) the business individuals with design technological knowhow to sort out the business challenges 5) the business individuals to construct the actual system 6) the actual functioning system developed from architectural products.

F6) Strengths and Weaknesses: ZF has been used to create FEAF, DoDAF and TOGAF and classifies architectural products model. It offers a series of views and abstractions and visualization support tools to organize enterprise activities. It identifies various unique stakeholders and provides means of building and reproducing system and enterprise architecture. It has a role for resource and reference structure analysis and is the most comprehensive of the EAFs. It supports traceability from requirements to implementation and is also a basis for reuse optimization.

However, ZF lacks high level abstraction to guide the design and has implicit process without stakeholders' viewpoint for architecture development. It has no decisions on future architecture outcomes and architecture need, as well as governance structure as lacks. ZF lacks a city plan that is suitable for EAF information and is not standard written resulting in lack of uniformity on artifacts. Relation between cells is not defined, and the effects on system component on functionalities and interactions are not addressed [31]. Conformity rules, prescriptions on design justification, quality requirements, architecture advancement and explicit system development process are lacking. Views are not created using order and hence does not follow top-down concept.

3.2.2.2 Department of Defense Architecture Framework (DoDAF)

F1) Philosophy: DoDAF offer guideline on structure breaking to develop products and it is used for enterprise, system, and system of systems architectures [26]. It acts as universal concept for creating systems in DoD to enable the comparison of created systems and supports goals transformation through universal guidance. It supports armed warfare operations, business functions and procedures. It also establishes information sharing architecture for managing data and decision making [3] [6]

F2) Dimensions of the framework: DoDAF is divided into data and presentation levels, and the data level contains data elements and the presentation level contains outcomes and aspects.

F3) Structure of the framework: DoDAF defines four perspectives [DoD Arc. 07]. All views illustrate the architecture designs, span, descriptions, and connections among the other views. While the operational view concentrates on the actions and tasks, the system view concentrates on the system and applications. The technical standards view concentrates on the guidelines, uniformity and controls. It describes products using Core Architecture Data Model (CADM) and architecture products [7]

F4) Artifacts: It describes a set of 26 outputs and 4 views [3][26] [7]. The views are valid to each product with names related to information they document. The 26 products hold details for the architecture from the prerequisite stage to the execution stage.

F5) Architecture development process: It has Architecture Development Process Model (ADPM) that has six steps namely: problem definition, operation and requirements, functions and organizations, information and operation elements and rules, interfaces analysis architecture data, and system performance [26].

F6) Strengths and Weaknesses: It describes outputs and procedures that guarantee uniformity and stress on data for architecture assessments. It provides a common set of items and requirements used by vendors' tools to supply software. It gives a universal group of objects, requirements and architecture elements; and addresses current and target architectures, as well as uses capabilities to measure architectures. It has no unique architecture development, no specialized analysis techniques and offers limited guidance quality attributes. It neither records on architecture rationale nor software evolution method. No information is available on architectural management; and no complete business guidance, financial and technical analysis. All view outputs do not show unique architecture aspects.

3.2.2.3 The Open Group Architecture Framework (TOGAF)

F1) Philosophy: TOGAF is a general architecture development method used for reviewing preparation, architecture completeness checking, etc [36]. It has a common EAF for use by industries to create, maintain and use architectures. It is a structure for planning, assessing and constructing architectures for organization [27].

F2) Dimensions of the framework: TOGAF is divided into four categories namely: business, application, data and technical architectures.

F3) Structure of the framework: TOGAF has Architecture Development Method (ADM) that describes the step-wise activities performed in creating EA [9]. It has a repository with components and tools defining components category outputs. TOGAF consists of business, application architecture, data architecture, technical architecture and communications media programs.

F4) Artifacts: TOFAF has ADM for developing and maintaining business technological structures. It has a repository for describing general or special structural design components and the solutions repository that illustrates actual components implementation. The asset repository has resources, guidelines, templates and background information [31].

F5) Architecture development process: TOGAF process has eight phases that are repeated [27]. The phases include considering vision to be achieved, creating a detailed baseline and target business structure, determining target information and functions; determining the communication media; evaluating implementation possibilities; identifying projects, and evaluating the business opportunity; sorting the projects identified into priority order; and creating architectural stipulations; and modifying the created structure.

F6) Strengths and Weaknesses: It has different levels of abstractions and describes final structure. It has governance structure and offers guidelines for making choice. It describes specific view development method and generates and stresses on basic and structural traceability [36]. It has unique outcomes, and allows modifications and addition of outcomes. It lacks specific products and can be used in different environments. TOGAF can bypass, combine, and transform stages as needed; and guides in the usage of standard classification. It aligns solutions to business using support tools and describes various knowledge bases [35]. It has virtual repository with architectures assets and provides guidance on decision making, resources, and principles. It supports architecture evolution, functions creation, and mismatch risk reduction between business and technology. And it generates value, discovers business transformation opportunities and documents descriptions functions creation.

It does not define generation of EA and implementation of flexibility, as well as offers limited description on document templates. It lacks architecture description structures, models, tools and lacks specific prescription group of EA outcomes. There is limited detail on planning, maintaining, and producing high-quality EA. It does not offer various virtual products and has no descriptions and interactions between products [36].

3.2.2.4 Federal Enterprise Architecture Framework (FEAF)

F1) Philosophy: It is a planning model for improving performance, identifying potential investment areas, supporting goal realization, and providing guidance on segmenting system structure [4]. It also establishes an integrated organization roadmap for achieving objectives and puts agencies' functions under single EA [11][20].

F2) Dimensions of the framework: It has two dimensional organizations. The rows depict perspectives, and the columns portray product abstraction entities, activities and locations.

F3) Structure of the framework: It consists of four tiers defined separately and includes business, data, application and technology structures [11]. The top two rows are the business structures like ZF and the models are in the third, fourth, and fifth rows. The rows represent perspectives and each have a unique goal. It has three columns that represent the what, how, and where respectively. The architectural descriptive representations are formed at the meeting points of perspectives and abstractions. It is organized also into 5 tiers where each tier is dealing with specific issue [4].

F4) Artifacts: It has the business allusion model that gives view of a government's various functions; the component allusion model that gives view of the system supporting business functionality; the technical allusion model that defines tools and standards for building systems; the data allusion model that defines data standard method; and the performance allusion model that defines standard for EA delivered value [11].

F5) Architecture development process: It has four phases that include architectural analysis for defining vision; architectural for defining desired architectural segment state; investment and funding strategy for funding project; and program management for executing projects management.

F6) Strengths and Weaknesses: It is the most complete compared with ZF, TOGAF and GEAF. It offers classification of products like ZF and provides architectural development process like TOGAF [3]. It provides a structure for developing, maintaining, and implementing operating environments. It has principles; and shows movement on vision, strategy and knowledge repository. It defines additional views, perspectives, use of methods, work products and tools. It has a common business definition language and transformation. It segments enterprise for reusability, has best standards practices and open standards for interoperability.

It has no quality prerequisites and has limitedly support on plan validation. It has no design support for modeling and organization standards. It has no presentation guidance on transformation processes, plan, and on stakeholders. It has no organization structures, security guidance, repository, a unit product specification development method.

3.2.2.5 Treasury Enterprise Architecture Framework (TEAF)

F1) Philosophy: It was developed from ZF, FEAF, and DoDAF to organize information, support business strategic direction, harness the integration of treasury offices, facilitate information sharing, common use of requirements and establish common EA structure [8]. It was to establish management foundation structures and assets; identify and achieve objectives; and support managers, business and technical planners [12].

F2) Dimensions of the framework: It has a two dimensional organization matrix having four views on the vertical and four perspectives on the horizontal.

F3) Structure of the framework: It describes views and perspectives comparable to ZF columns and rows [8]. It contains sixteen units, and four perspectives that resemble FEAF and ZF. Its builder combines the builder and subcontractor of ZF and the views match with the columns of FEAF. FEAF data structure matches with TEAF functioning and knowledge structures, and information view. The EA elements are defined with their interrelationships and the structure is built using a set of product guidelines.

F4) Artifacts: TEAF is divided into direction, description, and accomplishment, with products for each of these parts defined [12]. The repository stores all the information.

F5) Architecture development process: TEAF has a nine step development process: planning, analysis, design, implementation, project fusion, functions, assessments, organization and control, and technology advancements.

F6) Strengths and Weaknesses: It describes structures for documenting, modeling EA and creating structure that is aligned with FEAF models and DoDAF products. It defines new perspectives on stakeholder important areas, guides on transition, and offers target description principles. It relates EA to enterprise life cycle, determines enterprise activities issues, addresses configuration management, and delineates modeling techniques. It guides on architecture management roles and responsibilities, and addresses information security and assurance issues [26]. It has standards and investment development; and defines structures and offers governing principles [8]. It defines an EA repository without common department-wide format or mechanism for interchange of structural information [8].

3.3 Determining EAF comparison perspectives and aspects

Perspectives are the comparison criteria that focus on a specific interest on an object and are used to classify content of an object by categorizing them into groups that have related aspects. Aspects are the contents connected to the perspective and can be said to be the attributes of the criteria. Table 2 identifies the perspectives and aspects used in the previous comparison works introduced in Section 2. The last column contains perspectives and aspects, for example, [3] used three perspectives for comparison, i.e. P1, P2, and P3. The P1 perspective has the aspects P1.1, P1.2 and so on. In table 2, when the same perspective or aspect is repeated in the papers, its ID is repeatedly used. Table 2 is referenced in the subsequent sections by quoting their Id.

The chapter identifies several perspectives from past research papers as well as our own newly suggested perspectives to enhance accuracy of selecting EAF. It should be noted that the perspectives and aspects in italic are the ones we suggested hence no reference is indicated. The details on these new perspectives are covered in both sections 3.3 and 3.4.

Paper	**Comparison of perspectives/aspects**
P1. Goals [3] [31] [24]	P1.1 Architecture definition and understanding P1.2 Architecture development process P1.3 Architecture evolution support P1.4 Architecture analysis P1.5 Architecture models P1.6 Design rationale P1.7 Design tradeoff P1.8 Standardization P1.9 Architecture knowledge base P1.10 Architecture verifiability
P2. Inputs [3] [31]	P2.1 Business drivers P2.2 Technology inputs P2.3 Business requirements P2.4 Information system environments P3.5 Current architecture P2.6 Non functional requirements
P3. Outcomes [3] [31] [24]	P3.1 Business model P3.2 System model P3.3 Information model P3.4 Computation model P3.5 Software configuration model P3.6 Software processing model

Paper	**Comparison of perspectives/aspects**
	P3.7 Implementation model P3.8 Platforms P3.9 Non functional requirements design P3.10 Transition design P3.11 Design rationale
P4. Views	P4.1 Planner P4.2 Owner P4.3 Designer P4.4 Builder P4.5 Subcontractor P4.6 User
P5. Abstractions [22] [24]	P5.1 What P5.2 How P5.3 Where P5.4 Who P5.5 When P5.6 Why
P6. System development life cycle [22] [24]	P6.1 Planning P6.2 Analysis P6.3 Design P6.4 Implementation P6.5 Maintenance
P7. Guide [32] [27] [26]	P7.1 Meta model P7.2 Procedure model P7.3 Modeling technique P7.4 Role P7.5 Specification document P7.6 Process completeness P7.7 Maturity model P7.8 Reference model guidance P7.9 Practice guidance P7.10 Governance guidance P7.11 Partitioning guidance P7.12 Prescription catalog
	P7.13 Providing guidance architecture descriptions *P7.14 Product definitions* *P7.15 Architecture development* *P7.16 Information reference resources* *P7.17 Tool builders* *P7.18 compliant architecture views* *P7.19 EA development process* *P7.20 Transition strategy and plan* *P7.21 Product and repository issues*
P8. Quality [28] [24] [27]	P8.1 Alignment P8.2 Integration P8.3 Value creation P8.4 Change management P8.5 Compliance

Paper	**Comparison of perspectives/aspects**
	P8.6 Taxonomy completeness P8.7 Vendor neutrality P8.8 Time to value P8.9 Information availability P8.10 Business focus
P9. Miscellaneous [26][24] [31]	P9.1 Basic organizational approach and views P9.2 Integrated EA product specifications descriptions P9.3 EA strategic vision and goals P9.4 Architecture principles provision P9.5 Product for specifying standards P9.6 Security considerations in EA P9.7 Tool support P9.8 EA repository issues P9.9 Explicit specification P9.10 Architecture domain P9.11 Analytical approach P9.12 Stakeholder P9.13 Application domain development process P9.14 Conformance
P10. Requirements	*P10.1 Supporting methodology domain* *P10.2 Developing interface* *P14.3 Automating EA development and population* *P14.4 Extending and customizing* *P14.5 Analyzing and Manipulating* *P14.6 Providing repository* *P14.7 Deploying architecture* *P14.8 Costing and Vendor Supporting*
P11. Principles	*P11.1 Describing views for integrating enterprise products* *P11.2 Developing organization solutions* *P10.3 Developing physical plant* *P10.4 Developing non-physical plant* *P10.5 Enterprise focus* *P10.6 Performing standard assessments* *P10.7 Assessing products* *P10.8 Assessing new technologies* *P10.9 Tailoring model views* *P10.10 Developing standard profiles* *P10.11 Developing architecture descriptions* *P11.12 Evaluating objectives against requirements* *P10.13 Identifying enterprise problems* *P11.14 Architecture information Repository* *P10.15 Mapping of interfaces and services* *P10.16 Addressing migration issues*

Table 2. Perspectives and aspects compared

3.3.1 The significance and relationships between the selected perspectives and aspects

The numbering of the perspectives for existing and newly suggested perspectives in sections 3.3 and 3.4 is based on their importance in regard to system or architecture building. The perspectives focus on specific interest on an object in question with a set of aspects that are the contents connected to them. Requirements are what the enterprise architecture developed is supposed to satisfy and goals are what are expected to be achieved by the architecture developed. Inputs are to support problem solving so that the gaps in the current system can be identified and worked on to achieve the enterprise set requirements. Outcomes are the expected end results offered by the architecture developed. Data is gathered in order to focus on different stakeholders concerns and to arrive at a solution. More than one level of system abstraction may be needed to mange complexity of the scope of problem coverage.

A system development job is said to be done when quality attributes are fulfilled as stated at the beginning during requirements gathering. To develop a software system from the EA blue prints, we need system development life cycle that guides in the development process. To ensure proper use of the resources at our disposal for both architecture and system creation, we need clear guidelines on how to perform the tasks. Finally, we need the rules which must be adhered to by all concerned to avoid system failure. These rules create consensus and harmony in the created items and so must be followed as stated by the designer who defines the principle.

3.3.2 The significance of the selected perspectives

1) Goals: Specific goals must be identified so that they are accomplished. Identification of goals reveals the specific reasons, purpose or benefits of accomplishing the goals. Identified goals enable establishment of concrete criteria for measuring progress toward the attainment of each set goal and by measuring progress, keeping track of progress, reaching target dates, and realizing motivation gained from previous accomplishments so that future encouragements are possible for achieving future goals. Set goals enables prioritization and allow configuration of means to realize them; and create attitudes, abilities, skills, and financial capacity to accomplish them. Setting goals enable planning and setting of steps wisely, and establishing a time frame for performing the steps. Known goals make it possible to determine whether they are realistic by representing objectives that create willingness and ability to achieve them.

2) Inputs: Inputs of EA are the outline that integrates process and goals in a business enterprise to provide core components of the business process like strategic planning, organizational design, business process reengineering, and systems delivery. Inputs help to effectively exploit the intellectual capital, otherwise, intellectual capital will have no value and meaning without inputs.

3) Outputs: Outputs are like views and models that describe the existing environment and are used to understand the gaps that exist when planning for the future preferred environment. Outputs are like design guidelines and patterns that are used in the activities to govern IT task in the most efficient and shortest path to develop future preferred environment. Lack of output makes it difficult to determine the current status and the path to develop future environment as increases cost of doing so.

4) Views: Views are depictions of the complete architecture that are meaningful to concerned stakeholders in the system. Views allow communication and understanding of architecture by all stakeholders, as well as verification that the system will tackle the concerns.

5) Abstractions: Abstractions enforce a progressive decomposition and maintain traceability from the conceptual design of the architecture to the detailed implementation guidance. It allows the removal of complicating factors by incrementally refining the details to manage the enterprise complexity. Lack of abstractions will result in complex systems that are difficult to manage as risks of systems failure due to inability to efficiently and effectively understand and communicate among concern the stakeholders.

6) System development life cycle: Architecture delivery process allows implementation of a standardized design methodology that consists of well defined artifacts, processes, roles and responsibilities. Implicit process definition fastens design cycles, clears handoffs from organization to organization, and reduces costs.

7) Guide: Guiding process is intended to assist in defining, maintaining, and implementing EAs by providing a disciplined and thorough approach to EA life cycle management. The guide describes EA maintenance, implementation, oversight and control. This guidance is critically important and without it, it is highly unlikely that an organization can effectively produce a complete and enforceable EA for optimizing its systems for value.

8) Quality: Failure to develop good quality software may result in making changes to the software may increase costs in unexpected ways and IT can become unmanageable. Attaining quality attributes are the rightful indication of knowing whether job has been done or not.

9) Miscellaneous: This perspective contains a variety of aspects which cannot fit in other perspectives but are critical for comparing the EAFs. Therefore, it is up to the architect to select the related aspects from this perspective based on the intended usage to be focused on.

3.3.3 New perspectives and aspects

10) Requirements (Should be the top most in the list of perspectives): The requirements are the basis on which consideration of obtaining and applying an inclusive EAF representation begins. With no requirements needing solutions, there will be no motivation towards creating enterprise architecture and hence no need for search for EAF for use. When requirements are identified you are able to anticipate requirements that are critical for use in selecting an EAF. The requirements will enable the focus on the performance that is realized by use of EAF on the key operations required in supporting the EA development activity. Requirements lessen risk and equip an organization enabling it to adjust to unexpected changes, align business operations to evade their failures, and develop integrated business processes in the entire enterprise. It enables achievement of gains in combining information to reduce production, time and cost [33] [34].

11) Principles/rules (No.10 in the list of perspectives): Principles define the fundamental basic rules and guidelines for the use and deployment of all IT resources and assets in the entire enterprise. They reflect a level of consensus between the various elements of the enterprise, and build a foundation for future IT decisions making. Principles are vital decisions that are well-formed and should be explicit with lucid applicability scope [30]. It can be too costly to start enforcement of principles after failure of system due to ignorance. Therefore, each organization should have basic principles used in creating and maintaining the EA and developed system. The aspects are included in Table 2.

Note that the details of the formats used in EAFs survey, the survey results and the detailed explanation of aspects can be found in [1].

3.4 Selecting comparison scales

This section compares comparison scales and selects the best to use. Different comparison scales were used by the different authors. The scale should easily and clearly support visualization, decision making, documentation and clarification of the comparison results. Comparisons of comparison scales used are summarized in Table 3. Rating scale is the easiest way to select EAFs.

Comparison scales	Paper	Advantage	Disadvantage
Mapping other EAFs on the selected reference EAF	[22]	Graphical presentation is more easily understood than text.	Difficult to visualize and decide for several isolated mappings
Using notations to indicate EAF item supporting status	[31] [3]	Comparison items can be displayed on a diagram	Lack of weights makes it difficult to make decision.
Populating table with text on EAF status on item support	[26] [24]	More refined comparison details can be achieved	Complex to document and make decision for large comparisons
Rating the items based on how EAFs addresses it	[27]	Weights are given and so making decision is easy	Rate allocation is justified by EAF survey and related work.
Showing EAF item support status capabilities by legend	[32]	Easy to present and visualize the results	Not easy to make decision because of lack of weighs.
Narrating comparison results of EAF items addressing	[28]	Very refined information results can be achieved	It is very difficult to visualize results given in text form.
Indicating EAF item support status using Yes or No/Blank	[24] [22]	Easy to present, visualize and understand the results	Making decision is difficult because of lack of weights.

Table 3. Comparison of scaling schemes

4. Introducing enterprise architecture framework selection scheme

In this section, we propose EAFSS for selecting the best EAF for each usage listed in Table 5. Figure 2 shows the steps EAFSS uses in the EAFs selection process. The subtotal shows the degree to which EAF addresses each perspective. The total weight for each EAF is how it addresses the aspects. The totals in Table 6 are the scores for each EAF. Rate is the ranking of the EAFs based on the scores.

Enterprise Architecture Framework Selection Scheme

Step 1. Select a usage *U* from the Usages in Table 4, i.e. U1~U9

Step 2. Choose the perspectives and aspects relevant to *U* from Table 6.

Step 3. Assign weights to each aspect chosen in Step 2.

Step 4. Calculate the value of each EAF in Table 7 using the formula:

$$\text{Subtotal}_p = \sum_{i=1}^{n} (w_i \times r(a_i)) \text{ ---------------------------------- (F1)}$$

Step 5. Choose the EAF in Table 6 whose total value is the greatest.

Fig. 2. EAF Selection Example

Table 6 can compare any set of perspectives to select EAFs. We can shrink or expand the number of perspectives, aspects, or EAFs in Table 6. To use Table 7, the following steps need to be performed:

1) remove any unwanted perspectives;
2) add any needed perspectives missing in Table 7;
3) give weights to all the aspects in Table 7;
4) calculate each subtotal using the following formula:

Let $a_1, \ldots, a_n$ be the all the aspects of a perspective *p*, $r(\mathrm{a_i})$ the rate for the aspect a_i and w_i the weight for a_i. Then

$$\text{Subtotal}_p = \sum_{i=1}^{n} (w_i \times r(a_i)) \text{ ---------------------------------- (F1)}$$

5) sum up all the subtotals to get the grand total and
6) select the best EAF based on the rating totals quality of items that results

The EAFs detailed survey is found in [1]

5. Comparing of enterprise architecture frameworks

5.1 Identifying Usages and Related Perspectives

We identified EAFs usages from literature review for use in application of our proposed scheme. The usages are to be used in selecting the EAFs to be used in a specific EA design. Table 4 shows the usages and perspectives that EAFs should address satisfactorily.

5.1.1 Summary of usages and perspectives

Paper	Perspectives used	Usages
[3]	P1. Goals P2. Inputs P3. Outcomes	U1. Selecting the best EAFs for system or EA development. U2. Tailoring EAF for system or EA development
[26]	P7. Guide P9. Miscellaneous	U1. Selecting an appropriate EAF U2. Incorporating best practices from other EAFs

Paper	Perspectives used	Usages
[24]	P1. Goals P3. Outcomes P5. Abstractions P8. Quality P9. Miscellaneous	U1. Selecting an EAF that fits the task at hand U3. Identifying EAFs features for suitability U4. Identifying similarities between the EAFs
[22]	P4. Views P6. System development life cycle P5. Abstractions	U1. Selecting EAF by determining a best-fit of EAF
[32]	P7. Guide	U3. Determining EAFs capabilities in supporting EA U5. Identifying requirements for developing EA descriptions U6. Developing EA descriptions
[31]	P1. Goals P2. Inputs P3. Outcomes	U1. Selecting best EAF based on the status U2. Tailoring EAF to make sense for specific situation U7. Adapting EAF framework
[28]	P8. Quality	U8. Identifying relationship among the EAFs views U9. Determining EAFs usage growth
[27]	P7. Guide P8. Quality	U1. Selecting the best EAF U2. Developing EAF by blending

Table 4. Review Summary of usages Extracted from Comparisons of EAFs

5.1.2 Usages relevant perspectives and aspects of usages

Different perspectives satisfy different needs in system or EA development. Table 4 above shows the usages and their related perspectives that must be addressed satisfactorily by the EAF that is selected. Table 5 below is developed from Table 4. Letter “U” indicates usage. This table is very significant in that it tells the user the perspectives and aspects that must be satisfied by the EAF to be selected for each usage. The aspects relevant under each perspective and aspects will vary based on usage environment.

Usage	Perspectives and Aspects
U1. Selecting the best EAF for system or EA development	P1. Goals P2. Inputs P3. Outcomes P4. Views P5. Abstractions P6. System development life cycle P7. Guide P8. Quality P9. Miscellaneous P10. Requirements P11. Principles
U2. Tailoring EAF for system or EA development	P1. Goals P2. Inputs

Usage	Perspectives and Aspects
	P3. Outcomes P4. Views P5. Abstractions P6. System development life cycle P9. Miscellaneous *P9.4 Architecture principles* *P9.5 Product for specifying standards* *P9.6 Security considerations in EA* *P9.7 Tool support* *P9.8 EA repository issues* *P9.9 Explicit specification* *P9.11 Analytical approach* *P9.12 Stakeholder* *P9.13 Application domain development process* *P9.14 Conformance* P10. Requirements
U3. Identifying EAFs features for suitability	P1. Goals P2. Inputs P3. Outcomes P4. Views P5. Abstractions P6. System development life cycle P9. Miscellaneous *P9.6 Security considerations in EA* *P9.7 Tool support* *P9.8 EA repository issues* *P9.9 Explicit specification* *P9.11 Analytical approach* *P9.12 Stakeholder* *P9.13 Application domain development process* *P9.14 Conformance*
U4. Identifying similarities between the EAFs	P1. Goals P2. Inputs P3. Outcomes P4. Views P5. Abstractions P6. System development life cycle P7. Guide P8. Quality P9. Miscellaneous P11. Principles
U5. Identifying requirements for developing EA descriptions	P1. Goals P2. Inputs P3. Outcomes

Usage	**Perspectives and Aspects**
	P5. Abstractions P10. Requirements
U6. Developing EA descriptions	P1. Goals P2. Inputs P3. Outcomes P4. Views P5. Abstractions P7. Guide P8. Quality P9. Miscellaneous *P9.6 Security considerations in EA* *P9.7 Tool support* *P9.8 EA repository issues* *P9.10 Architecture domain* *P9.12 Stakeholder* *P10. Requirement*
U7. Adapting an EAF	Goals P1.2 Architecture development process P1.5 Architecture assessment P1.5 Architecture models Inputs P2.1 Business drivers P2.3 Business requirements P3. Outcomes P3.1 The business model P3.2 System model P3.3 The information model P3.4 The computation model P3.6 The processing model P4. Views P4.1 The scope P4.2 The owner P4.3 The designer P4.4 The builder P5. Abstractions P5.1 The what P5.2 The how P5.3 The where P5.4 The who P5.5 The when P5.6 The why P8. Quality P8.1 Alignment P8.2 Integration

Usage	**Perspectives and Aspects**
	P8.3 Value creation P8.6 Taxonomy completeness P8.10 Business focus P9. Miscellaneous *P9.1 Organizational approach and views* P9.3 EA strategic vision and goals P9.7 Tool support P9.10 Analytical approach P9.11Stakeholder
U8. Identifying relationship among the EAFs views	P1. Goals P2. Inputs P3. Outcomes P4. Views P5. Abstractions P10. Requirements
U9. Determining EAFs usage growth	P7. Guide P8. Quality P9. Miscellaneous *P9.3 EA strategic vision and goals* P9.4 Architecture principles *P9.6 Security considerations in EA* *P9.7 Tool support* *P9.8 EA repository issues* *P9.9 Explicit specification* *P9.11 Analytical approach* *P9.12 Stakeholder* *P9.13 Application domain development process* *P9.14 Conformance* P11. Principles

Table 5. Relating Perspectives and Aspects to Usages

U1. Selecting the best EAF for system or EA development: Failures have been evidenced in the past in systems developed using traditional methods [27]. It is like trying to put up a complex construction with no plan. Of course the outcome will not be pleasing and most likely it will be a failure. Likewise, selecting the right EAF based on requirements is one very crucial step in EA development that guarantees creation of the right EA that aligns systems to IT and business needs. This result in a successful system that is build from the right plan developed using the right EAF.

To select appropriate EAF for system or EA development, we have to consider all the perspectives to determine how they are addressed by the EAFs. To start with, the problem must be identified and related requirements gathered towards the solution. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure a complete solution to the problem. Views must be identified that focus on different stakeholders affected by the problem. Based on problem complexity more than one model may be needed and a model may need several levels of abstractions to manage complexity and to enhance understanding and communication among the stakeholders.

Qualities of the products of an EAF that are used to develop system or EA are vital for their proper and correct use. To develop a software system from the EA blue prints, we need system development life cycle that guides the development process. Proper use of the resources for both architecture and system creation requires guidelines on how tasks are performed. Rules/principles are needed that create consensus and harmony in creating and maintaining system or EA. There is need for other additional aspects that are critical for enhancing system or EA effort, which are selected from the miscellaneous perspective.

U2. Tailoring EAF for system or EA development: EAFs are not complete as literature tells and hence no single one in isolation can offer complete solution [31] [12] [19] [3]. Therefore, tailoring becomes a necessity for any selected EAF to enhance EA effort. An EAF may cover most parts of definition but combining them in application can increasingly enhance universal completeness. Also, incorporating best practices from other EAFs and applying them improves the capability of the selected EAF in supporting the building of EA.

To tailor EAF for system or EA development, we have to consider several perspectives that must be fulfilled by the EAF to be selected. To start with, the problem must be identified and related requirements gathered towards its solution. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure a complete solution to the problem. Views are needed that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed that may need several levels of abstractions to manage complexity and enhance understanding and communication among the stakeholders.

To develop a software system from the EA blue prints, we need system development life cycle that guides the development process. There is need for other additional aspects that are critical for enhancing system or EA effort, which are selected from the miscellaneous perspective.

U3. Identifying EAFs features for suitability: Although EAF have many features that are all important at one point in the use of EAF, others are more critical and when they miss they render the EAF unsuitable for use. These very critical features should be identified so that EAFs can be subjected to assessment using them to determine their suitability for application in the different usages.

To identify EAF features for system or EA development, we have to consider several perspectives that must be fulfilled by the EAF to be selected. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure a complete solution to the problem. Views are needed that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed that may need several levels of abstractions to manage complexity and enhance understanding and communication among the stakeholders.

To develop a software system from the EA blue prints, we need system development life cycle that guides the development process. There is need for other additional aspects that are critical for enhancing system or EA effort, which are selected from the miscellaneous perspective.

U4. Identifying similarities between the EAFs: If the similarities in the EAFs can be known, the selection process of the EAF can be reduced tremendously. This is because assessments will only be done on the dissimilarities by assuming that the others are the same. A lot of resource and effort will be saved by lessening the EAF assessment process.

To identify similarities in the different EAFs, we have to consider several perspectives that must be fulfilled by the EAF to be selected. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled. Outcomes must be known to ensure a complete solution to the problem. Views are needed that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed that may require several levels of abstractions to manage complexity and enhance understanding and communication among the stakeholders.

Qualities of the products of an EAF that are used to develop system or EA are vital for their proper and correct use. To develop a software system from the EA blue prints, we need system development life cycle that guides the development process. Proper use of the resources for both architecture and system creation require guidelines on how tasks are performed. Rules/principles are needed that create consensus and harmony in creating and maintaining system or EA. There is need for other additional aspects that are critical for enhancing system or EA effort, which are selected from the miscellaneous perspective.

U5. Identifying requirements for developing EA descriptions: Requirements are one of the big problems that most organizations face because they keep on changing with time. If we can imagine of a case where all requirements are identified and set aside for sharing, a lot of problems related to changes can really be reduced. Shared requirements throughout the organization avoid redundancy and system simplifies maintenance process.

In order to identify requirements for developing EA using EAF for system or EA development, we should consider several perspectives. The problem must be identified and related requirements gathered towards its solution. Goals should be set from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure a complete solution to the problem. Views are needed that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed and hence the need for several levels of abstractions.

U6. Developing EA descriptions: EAF are to support the creation of EA which includes the products and their descriptions. Therefore, it is very critical to consider EAF that is capable of creating EA descriptions that covers all stakeholders that have concerns in the system to be developed. Of course, each EAF has products but the extent of coverage in regard to enterprise activities differs and so it is of concern to know which one supports best the development of the needed EA descriptions.

To develop EA descriptions using EAF, we have to consider several perspectives that must be fulfilled by the EAF to be selected. To start with, the problem must be identified and related requirements gathered towards its solution. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure a complete solution to the problem. Views are needed that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed that may require several levels of abstractions to manage complexity and enhance understanding and communication among the stakeholders.

Qualities of the products of an EAF that are used to develop system or EA are vital for their proper and correct use. Proper use of the resources for both architecture and system creation requires guidelines on how tasks are performed. There is need for other additional aspects that are critical for enhancing system or EA effort, which are selected from the miscellaneous perspective.

U7. Adapting an EAF: The task of incorporating best practices from the other EAFs into the best EAF selected can be lessened by adapting an EAF. We have EAFs that can universally be adapted to various requirements. These are the EAFs that are not developed for specific domain. Therefore, time, effort and cost is reduced by avoiding tailoring when adaptation becomes the best option in some situations. Knowing which EAF is best for adapting is of great importance.

To adapt EAF for system or EA development, we have to consider several perspectives that must be fulfilled by the EAF to be selected. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure that complete solution components to the problem are included. Views are needed so that selection is made on views that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed that may require several levels of abstractions to manage complexity and enhance communication among the stakeholders.

Qualities of the products of an EAF that are used to develop system or EA are vital for their proper and correct use. There is need for other additional aspects that are critical for enhancing system or EA effort, which are selected from the miscellaneous perspective.

U8. Identifying relationship among the EAF views: Views are meant to focus on the different stakeholders and to make communication between them possible and best. Lack of enough views means that some stakeholders may not be addressed and the system will not reflect the true status of the enterprise. Therefore, it is critical to identify and relate the views to know the impact of missing them in the EAFs.

To adapt EAF for system or EA development, we have to consider several perspectives that must be fulfilled by the EAF to be selected. To start with, the problem must be identified and related requirements gathered towards its solution. A set of goals is needed from which we select what is relevant in solving the problem. Existing inputs to support in solving the problem needs to be identified to point out the gaps to be filled to satisfy the requirements. Outcomes must be known to ensure a complete solution to the problem. Views are needed that focus on different stakeholders that are affected by the problem. Based on problem complexity, more than one model may be needed that may require several levels of abstractions to manage complexity and enhance understanding and communication among the stakeholders.

U9. Determining EAFs usage growth: As time goes by, systems become more and more complex and managing them without employing EAFs become almost impossible. This is evidenced by the many failures of organization systems which are developed without using EAFs. However, as organizations become aware of the benefits of using EAFs, more and more organizations are motivated to use the EAF, thus the growth in EAFs usage is increasing with time. It is vital to know the EAF usage growth so that organizations are encouraged to use it.

To identify EAF growth for system or EA development, we have to consider several perspectives that may encourage the growth in EAF usage. EAF value creation and benefits through IT infrastructure encourages organizations to adopt it. This calls for addressing quality of items those results in valuable EA development. Proper use of the resources for both EA and system creation reduces cost and time, hence organizations are motivated to use EAF, however, guidelines on how to perform tasks should be available. Additional aspects are needed to enhance system or EA effort, which are selected from the miscellaneous perspectives.

5.2 Comparing selected EAFs

EAFs must be compared because their applications concepts, strengths and weaknesses differ. The user must decide the EAFs based on needs and the paper only guides the user. The weight indicates the degree to which an aspect is addressed. The paper user needs to survey the EAFs, so as to rate and assign appropriate weights to aspects. EAFs differ in concepts and no one is best for any organization. We used perspectives and aspects that we believe are critical in comparing and evaluating EAFs for use. They may not all be relevant to an organization and some may be more critical than others based on needs. We provided a cornerstone to start EAFs evaluation process. Weights are subject to change, and our rates which are fixed may not all be approved because EAFs are continuously being improved. At the end of the comparison process, we should understand the strengths and weaknesses of each EAFs based on needs. As shown in Table 6, “0” indicates that the EAF does not entirely address an aspect. “1” indicates that the EAF does not adequately address an aspect. “2” indicates that the EAF does address an aspect adequately. Using rates like 1 and 4 leaves a big gap that may not give reflective results. Close range of rates gives better results because thorough study is needed for correct EAFs rating. Due to lack of space we used three rates instead of five e.g. 0~4.

Comparison perspectives/Aspects	**ZF**	**DoDAF**	**TOGAF**	**FEAF**	**TEAF**
P1. Goals					
P1.1 Architecture definition and understanding	1	2	2	2	2
P1.2 The architecture development process	1	2	2	2	2
P1.3 Architecture transition strategy	0	2	2	1	0
P1.4 Architecture evolution support	0	2	2	2	2
P1.5 Architecture assessment	2	2	2	2	1
P1.5Architecture models	2	2	2	2	2
P1.6 Design tradeoffs	1	2	1	1	1
P1.7 Design rationale	1	1	2	1	1
P1.8 Standardization	0	2	2	1	2
P1.9 The architecture knowledge base	0	2	2	2	2
P1.10 Architecture verifiability	0	0	2	0	0
Subtotal	**8**	**19**	**21**	**16**	**15**
P2. Inputs					
P2.1 Business drivers	1	2	2	2	2
P2.2 Technology inputs	0	2	2	2	2
P2.3 Business requirements	2	2	2	2	2
P2.4 Information system environment	1	2	2	2	2
P2.5 Current architecture	1	2	2	2	2
P2.6 Quality requirements	1	1	2	1	0
Subtotal	**6**	**11**	**12**	**11**	**10**
P3. Outcomes					
P3.1 The business model	2	2	2	2	2
P3.2 System model	2	2	2	2	2
P3.3 The information model	2	2	2	2	2
P3.4 The computation model	2	2	2	2	2
P3.5 The software configuration model	0	0	2	0	0
P3.6 The processing model	2	2	2	2	2
P3.7 The implementation model	1	2	2	1	1
P3.8 The platform	2	2	2	2	2
P3.9 The quality design	1	1	2	1	0
P3.10 The transition design	0	2	2	2	0
P3.11 The design rationale	0	1	1	0	2
Subtotal	**14**	**18**	**21**	**16**	**15**
P4. Views					
P4.1 The scope	2	2	0	2	2
P4.2 The owner	2	2	2	2	2
P4.3 The designer	2	2	2	2	2
P4.4 The builder	2	2		2	2
P4.5 The subcontractor	2	0	0	2	
P4.6 The user	2	0	0	0	0
Subtotal	**12**	**8**	**4**	**10**	**8**

Comparison perspectives/Aspects	ZF	DoDAF	TOGAF	FEAF	TEAF
P5. Abstractions					
P5.1 The what	2	2	0	2	2
P5.2 The how	2	2	1	2	2
P5.3 The where	2	2	0	2	2
P5.4 The who	2	2	1	0	0
P5.5 The when	2	0	0	0	0
P5.6 The why	2	0	0	0	0
Subtotal	**12**	**8**	**2**	**6**	**6**
P6. System development life cycle					
P6.1 Domain identification	1	2	1	2	2
P6.2 Planning	1	2	0	2	2
P6.3 Analysis	1	2	1	2	1
P6.4 Design	1	2	1	2	2
P6.5 Implementation	1	1	1	2	2
P6.6 Maintenance	0	0	1	1	0
Subtotal	**5**	**9**	**5**	**11**	**9**
P7. Guide					
P7.1 Meta model	1	1	0	0	1
P7.2 Procedure model	1	2	2	2	2
P7.3 Modeling Technique	0	2	1	0	1
P7.4 Role	0	1	0	2	2
P7.5 Specification document	2	2	1	2	2
P7.6 Process completeness	1	1	2	1	1
P7.7 Maturity model	0	1	1	1	1
P7.8 Reference model guidance	0	2	1	2	1
P7.9 Practice guidance	0	2	1	1	1
P7.10 Governance guidance	0	0	1	1	1
P7.11 Partitioning guidance	1	1	1	2	1
P7.12 Prescription catalog	1	1	1	2	2
P7.13 Providing guidance on architecture descriptions	*1*	*2*	*2*	*1*	*1*
P7.14 Product definitions	*1*	*2*	*1*	*1*	*1*
P7.15 Architecture development	*2*	*1*	*2*	*0*	*0*
P7.16 Information reference resources	*2*	*0*	*2*	*0*	*0*
P7.17 Tool builders	*1*	*1*	*2*	*1*	*0*
P7.18 compliant architecture views	*0*	*2*	*2*	*0*	*2*
P7.19 EA development process	1	2	2	1	1
P7.20 Transition strategy and plan	0	1	2	1	2
P7.21 Product and repository issues	1	1	0	1	1
Subtotal	**16**	**28**	**27**	**21**	**24**
P8. Quality					
P8.1 Alignment	2	2	2	2	2
P8.2 Integration	2	1	2	2	2
P8.3 Value creation	2	0	2	2	0
P8.4 Change management	2	2	2	1	0

Comparison perspectives/Aspects	**ZF**	**DoDAF**	**TOGAF**	**FEAF**	**TEAF**
P8.5 Compliance	1	2	2	2	2
P8.6 Taxonomy completeness	2	1	1	1	1
P8.7 Vendor neutrality	2	2	1	2	1
P8.8 Time to value	1	1	1	2	1
P8.9 Information availability	1	2	2	1	1
P8.10 Business focus	2	1	1	2	2
P8.11 EA focus	2	2	1	2	2
P8.12 Explicit detail of products	1	2	1	1	1
Subtotal	**20**	**18**	**18**	**20**	**15**
P9. Miscellaneous					
P9.1 Basic organizational approach and views	2	1	1	1	1
P9.2 Integrated EA product specifications descriptions	1	2	0	1	1
P9.3 EA strategic vision and goals	2	1	2	2	2
P9.4 Architecture principles	0	1	2	1	1
P9.5 Product specifying standards	0	1	2	0	2
P9.6 EA security issues	0	1	1	0	1
P9.7 Tool support	2	1	1	1	1
P9.8 EA repository issues	0	2	0	1	1
P9.8 Explicit specification	1	2	2	2	1
P9.9 Architecture domain	2	2	1	1	1
P9.10 Analytical approach	2	0	0	2	2
P9.11Stakeholder	2	1	0	2	2
P9.12 Application domain development process	1	2	1	2	2
P9.13 Conformance	0	1	2	1	1
Subtotal	**15**	**18**	**15**	**17**	**19**
P10. Requirements	*0*	*0*	*0*	*0*	*0*
P14.1 Methodology domain supporting	*2*	*2*	*1*	*2*	*2*
P14.2 Developing interface	*1*	*2*	*2*	*1*	*1*
P14.3 Automating EA development and population	*2*	*2*	*2*	*0*	*0*
P14.4 Extending and customizing	*1*	*1*	*2*	*0*	*0*
P14.5 Analyzing and Manipulating	*2*	*2*	*2*	*1*	*1*
P14.6 Providing repository	*0*	*2*	*1*	*0*	*1*
P14.7 Costing and Vendor Supporting	*1*	*2*	*2*	*0*	*0*
Subtotal	***9***	***13***	***12***	***4***	***5***
P11. Principles					
P11.1 Integrating enterprise systems	*2*	*2*	*1*	*2*	*1*
P11.2 Developing organization solutions	*2*	*2*	*1*	*1*	*1*
P11.3 Developing physical plant	*2*	*2*	*2*	*0*	*1*
P11.4 Developing non-physical plant	*1*	*1*	*1*	*2*	*1*
P11.5 Enterprise focus	*2*	*2*	*1*	*1*	*2*
P11.6 Performing standard assessments	*0*	*2*	*2*	*1*	*1*
P11.7 Assessing products	*2*	*0*	*2*	*1*	*1*

Comparison perspectives/Aspects	ZF	DoDAF	TOGAF	FEAF	TEAF
P11.8 Assessing new technologies	*1*	*2*	*2*	*1*	*1*
P11.9 Tailoring model views	*2*	*1*	*2*	*0*	*0*
P11.10 Developing standard profiles	*0*	*2*	*1*	*0*	*0*
P11.11 Developing architecture descriptions	*0*	*0*	*2*	*0*	*0*
P11.12 Evaluating objectives against requirements	*2*	*1*	*2*	*1*	*1*
P11.13 Determining enterprise problems	*2*	*2*	*1*	*2*	*2*
P11.14 Architecture information Repository	*2*	*2*	*1*	*0*	*1*
P11.15 Mapping of interfaces and services	*0*	*2*	*0*	*2*	*0*
P11.16 Addressing migration issues	*0*	*0*	*2*	*0*	*2*
Subtotal	**20**	**23**	**23**	***14***	***15***
Total	**138**	**173**	**158**	**147**	**141**

Not addressed: 0; Partially addressed: 1; Fully addressed: 2

Table 6. Comparison of EAFs

The perspectives used in Table 6 may not be all applicable in all cases; and some may be more critical than the others as dictated by requirements. However, this table serves as a basis on which individual EAF selection evaluation can begin. Comparison of all perspectives is complete in Table 6. Table 6 can compare any set of perspectives to select EAFs. We can shrink or expand the number of perspectives, aspects, or EAFs in Table 6. To use Table 7, the following steps need to be performed:

1) remove any unwanted perspectives;
2) add any needed perspectives missing in Table 7;
3) give weights to all the aspects in Table 7;
4) calculate each subtotal using the following formula:
   Let a_1, …, a_n be the all the aspects of a perspective p, $r(\mathrm{a_i})$ the rate for the aspect a_i and w_i the weight for a_i. Then

$$\text{Subtotal}_p = \sum_{\mathrm{i}=1}^{\mathrm{n}} (w_{\mathrm{i}} \mathrm{x} r(\mathrm{a}_i)) \tag{F1}$$

5) sum up all the subtotals to get the grand total and
6) select the best EAF based on the rating totals.

Perspectives and aspects serve as a basis on which individual EAF selection assessment can begin. The weight value is how the EAF addresses the aspect. The EAFs survey is found in [1], the result is not presented due to space. All the perspectives and aspects in Table 6 are assumed to have equal importance. Under that assumption, DoDAF leads. However, for specific usages DoDAF may not be the best.

5.3 Comparison results

In Table 6, all the perspectives and aspects were presented as if they have equal importance. In this case, it can be seen from Table 6 that DoDAF is leading. However, there is no consideration made for a specific usage and DoDAF may not be the best when we consider each usage at a time for us to compare the EAFs for selecting the best for that specific usage. Based on the usage chosen by an organization, the architects are to conduct comparison using the relevant perspectives and aspects given in Table 6 and adjust them to fit the situation.

Table 6 can shrink or expand in number of perspectives, aspects and EAFs. The total rate per perspective shows how close or diverse the EAFs are in addressing aspects for each perspective. The closer the rates the better because this shows that there is significance commonality in EAF consideration of those perspectives. The bigger the gap in rates for EAF the worse it is because this shows the commonality is lacking in those perspectives in regard to EAFs.

6. Examples of EAFSS application

In this section, we propose an Enterprise Architecture Framework Selection Scheme (EAFSS) that can be used to identify the best EAF for each usage listed in Table 5. Figure 2 shows the steps of the EAFSS.

Classification of EAFs based on perspectives addressing is possible, and this determines what the different EAFs support in regard to EA. It is possible to assess EAF support for specific usage and its worth towards solving the problem, by ensuring that the selected EAF offers the best solution to the organization. Therefore, in this section we demonstrate how the usages and perspectives can be applied to select the best fit EAF for different usages.

Enterprise Architecture Framework Selection Scheme

Step 1. Select a usage *U* from the Usages in Table 8, i.e. U1~U9

Step.2. Choose the perspectives and aspects relevant to *U* from Table 6.

Step 3. Assign weights to each aspect chosen in Step 2.

Step 4. Calculate the value of each EAF in Table 7 using the formula:

$$\text{Subtotal}_p = \sum_{i=1}^{n} (w_i \times r(a_i))$$

Step 5. Choose the EAF whose value is the greatest.

Figure 3. The EAFSS Steps

EAF is just a planned outline that can be tailored by any organization for a tactical target described by concerned stakeholders. Specifically, EAFs do not indicate the number of levels to model decomposition required to achieve quality of information sufficient for specific objective. The definition of a process to model and guiding principle to abide by are generally not given.

The information we presented in this chapter on EAFs can be used to adapt some of them to different organizations with different needs because information reveals some generality in them. Combination of elements of different EAFs in Table 7 can be performed to satisfy unique development requirements. This enhances speed and accuracy of decision making on development requirements. Incorporation of best practices from other EAFs is performed to develop an integrated EAF that enhances EA effort. Our approach allows the harmonization of tradeoff and recommendation, prior to selecting EAF, by considering EA effort based on the problem to solve, or information needed to solve the problem.

For tailoring of FAFSS, the following tasks need be performed:

1. The user can revise the list of perspectives and aspects based on requirements.
2. The user can give the weights to perspectives aspects according his satisfaction.
3. The user can rate the EAFs based on the allocated weights.

6.1 EAFSS application examples

Here we show two different examples for validating the application of our EAFSS. The first example is the usage for selecting the best EAF for system or EA development, and in this example we considered all the perspectives and aspects in Table 6. This is because for such usage our EAFSS requires to consider all the perspectives and aspects of our selection process. The second example shows how to select the best EAF for determining the EAF usage growth, and in this example we considered relevant certain perspectives and only certain aspects of those relevant considered perspectives.

6.1.1 Selecting the best EAF for system or EA development

This is the usage U1 in Table 5. From Table 6, we can see that all the perspectives are relevant. For this example, we assume w = 1 for all the aspects in the formula (F1) in Section 4.2. This means that all the perspectives and all the aspects are considered to be of equal importance to the assessment and selection of EAF for system or EA development. Therefore, we just add the subtotals to get the total for each EAF and select the best EAF based on the results as shown in Table 7. According to the results in Table 7 DoDAF is the best EAF for U1.

Comparison perspectives/Aspects	ZF	DoDAF	TOGAF	FEAF	TEAF
P1. Goals					
Subtotal	8	19	21	16	15
P2. Inputs					
Subtotal	6	11	12	11	10
P3. Outcomes					
Subtotal	14	18	21	16	15
P4. Views					
Subtotal	12	8	4	10	8
P5. Abstractions					
Subtotal	12	8	2	6	6
P6. System development life cycle					
Subtotal	5	9	5	11	9
P7. Guide					
Subtotal	16	28	27	21	24
P8. Quality					
Subtotal	20	18	18	20	15
P9. Miscellaneous					
Subtotal	15	18	15	17	19
P10. Requirements					
Subtotal	9	13	12	4	5
P11. Principles					
Subtotal	20	23	23	14	15
Total	**138**	**173**	**158**	**147**	**141**

Table 7. Selecting EAF for system or EA development

6.1.2 Adapting an EAF as a tailoring example of EAFSS

For this example we are using U7 and the relevant perspectives to this usage from Table 4 are P1, P2, P3, P4, P5, P8, and P9. We used these seven perspectives with selected relevant aspects only as shown in Table 8.

Let i be the set of all the aspects of a perspectives such that $i=i_1 \cup i_2$

- i_1 : set of relevant aspects
- i_2 : set of irrelevant aspects (are ignored because they have a value of zero)

Furthermore the relevant aspects are given weights according to their importance and relevance. The weights are shown in parentheses in the leftmost column of Table 8. For this example, a rating of 1~4 was used.

We are considering an aspect "*a*" and applying weights based on the priority "a" is given in regard to the usage. *For irrelevant aspects their weights are automatically 0 without saying it.*

- $a \in i_1 \Rightarrow w_i = 1{\sim}4$
- $a \in i_2 \Rightarrow w_i = 0$

The column with WR for weighted rate contains the results of multiplying the weight and the rate of each aspect. Now in Table 8 each subtotal for a perspective is obtained by adding the WRs for the aspects of the perspective and by summing up all the subtotals we get the score for each EAF. The best EAF for the usage is the one with the highest score. In this example ZF is the best choice.

Comparison perspectives/Aspects	**ZF**		**DoDAF**		**TOGAF**		**FEAF**		**TEAF**	
	Rate	WR	Rate	WR	Rate	WR	Rate	WR	Rate	WR
P1. Goals										
P1.2 Architecture development process (2)	1	2	2	4	2	4	2	4	2	4
P1.5 Architecture assessment (3)	2	6	2	4	2	4	2	4	1	2
P1.5 Architecture models (4)	2	8	2	4	2	4	2	4	2	4
P2. Inputs										
P2.1 Business drivers (3)	1	3	2	6	2	6	2	6	2	6
P2.3 Business requirements (3)	2	6	2	6	2	6	2	6	2	6
P3. Outcomes										
P3.1 The business model (3)	2	6	2	6	2	6	2	6	2	6
P3.2 System model (3)	2	6	2	6	2	6	2	6	2	6
P3.3 The information model (3)	2	6	2	6	2	6	2	6	2	6

Comparison perspectives/Aspects	ZF		DoDAF		TOGAF		FEAF		TEAF	
P3.4 The computation model (3)	2	6	2	6	2	6	2	6	2	6
P3.6 The processing model (3)	2	6	2	6	2	6	2	6	2	6
P4. Views										
P4.1 The scope (2)	2	4	2	4	0	0	2	4	2	4
P4.2 The owner (3)	2	6	2	6	2	6	2	6	2	6
P4.3 The designer (4)	2	8	2	8	2	8	2	8	2	8
P4.4 The builder (4)	2	8	2	8			2	8	2	8
P5. Abstractions										
P5.1 The what (3)	2	6	2	6	0	0	2	6	2	6
P5.2 The how (3)	2	6	2	6	1	3	2	6	2	6
P5.3 The where (2)	2	4	2	4	0	0	2	4	2	4
P5.4 The who (1)	2	2	2	2	1	1	0	0	0	0
P5.5 The when (2)	2	4	0	0	0	0	0	0	0	0
P5.6 The why (3)	2	6	0	0	0	0	0	0	0	0
P8. Quality										
P8.1 Alignment (4)	2	8	2	8	2	8	2	8	2	8
P8.2 Integration (2)	2	4	1	2	2	4	2	4	2	4
P8.3 Value creation (4)	2	8	0	0	2	8	2	8	0	0
P8.6 Taxonomy completeness (4)	2	8	1	4	1	4	1	4	1	4
P8.10 Business focus (4)	2	8	1	4	1	4	2	8	2	8
P9. Miscellaneous										
P9.1 Organizational approach and views (3)	2	6	1	3	1	3	1	3	1	3
P9.3 EA strategic vision and goals (3)	2	6	1	3	2	6	2	6	2	6
P9.7 Tool support (2)	2	4	1	2	1	2	1	2	1	2
P9.10 Analytical approach (3)	2	6	0	0	0	0	2	6	2	6
P9.11Stakeholder (4)	2	8	1	4	0	0	2	8	2	8
Totals		**175**		**128**		**111**		**153**		**143**

Table 8. Adapting EAF

7. Conclusions

There are a number of already established EAF in use today; some of these EAFs were developed for use in very specific areas, whereas others have broader functionality. In this chapter a review of these EAFs is presented. The chapter conducted a comparative analysis on the basis of previous research. Further, the chapter discussed the importance and background of the selected EAFs. Therefore, the chapter provides a comparison of several EAFs that can then be used for guidance in the selection of an EAF that meets the needed criteria.

The chapter has covered a broad introduction to the field of EAF. As I have shown from the review of EAFs conducted, these methodologies are quite different from each other, both in goals and in approach. This is good and bad news. It is bad news, in that it increases the difficulty for organizations in choosing one single EAF methodology. It is difficult to choose between methodologies that have very little in common. The good news is that these EAFs methodologies can be seen as complementing each other. For many organizations, the best choice is all of these EAFs, blended together in a way that works well within that organization's constraints. The chapter provides a scheme that blends together all the compared EAFs. Therefore, the chapter provides a good starting place for understanding the value of each of these EAFs and how they can complement each other to bring the business side and the technology sides together, so that both can work effectively toward the same goals.

Either route one chooses, we should remember that EAF is a path, and not a destination. EAF has no value unless it delivers real business value in the shortest time possible. The most important goal of any EAF is to bring the business side and the technology sides together, so that both are working effectively toward common goals. In many organizations, there is a culture of distrust between the technology and business folks. There is no EAF methodology that can bridge the divide unless there is a real commitment to change. That commitment must come from the uppermost level of the organization. Methodologies cannot solve people problems; they can only provide a framework in which those problems can be solved. However, as soon as one has that commitment to change, an EAF methodology can be a valuable tool for guiding that change. This change can manifest itself in many ways. The many ways include improvements in using IT to drive business adaptability; closer partnership between business and IT groups; improved focus on organizational goals; improved morale, as more individuals see a direct correlation between their work and the organization's success; reduced numbers of failed IT systems; reduced complexity of existing IT systems; improved agility of new IT systems; and closer alignment between IT deliverables and business requirements.

Due to failures that have been reported in the past as indicated in this chapter [27], it is true that an organization that does well in these key areas will be more successful than the one that doesn't. This is true regardless of whether success is measured with tangibles, such as profitability and return on investment, or intangibles, such as customer satisfaction and employee turnover. The starting point for any EA is some critical self-analysis. These are some of the questions you should ask yourself during the analysis: does your organization spend too much money building IT systems that deliver inadequate business value? Is IT seen as improving or hampering business agility? Is there a growing divide between your business and IT folks? And, lastly, perhaps the most important question of all: is your organization truly committed to solving these problems, and does that commitment come from the highest levels of the organization? If the answer to all of these questions is "yes," EA is your path. It's up to you to take that next step.

Enterprise architects are still stuck in the technology weeds and the resulting lack of connection with business leadership. One cannot make a business case for EA if we do not connect with the business aspects of it. Furthermore, let us highlight the futility of comparing EAFs, because one can admire and maybe express a preference for the architectures of various buildings, however, can we really "compare" EAF in a meaningful way? This is a challenge to us all and that's why for many organizations, the best choice is all of these EAFs, blended together in a way that works well within that organization's constraints. The generic scheme is the best option because it borrows from the other EAFs.

We believe that our approach is better than any of the past approaches. This is supported by the unique information provided on EAFs usages that has not been considered by the past researches. Our scheme can support organizations to systematically select the best EAFs for the usages they envisage to design a desired architecture. The scheme is the best because it borrows perspectives and aspects from all of the major existing EAFs, and blends them together in a way that works well within any organization's constraints. It is the duty of the organization to decide what to include in their blending that best fits the organization constraints.

Tailoring of the selected EAF: Tailoring of selected EAF is agreeable and anticipated because almost all of existing EAFs are not complete to offer support to EA design in isolation. Tailoring is the process of borrowing best practices to fill the gaps identified in the best EAFs based on comparison results. For example, DoDAF is leading in Table 7, however, it is not supporting all the usages and to enhance the EA design effort, tailoring and integration of best practices must be conducted.

8. References

[1] Agnes Owuato Odongo, "E-government system architecture design guided by an e-government development process and an enterprise architecture framework," Master of Science in Engineering Thesis, KAIST, Korea, 2009.

[2] A Practical Guide to Federal Enterprise Architecture, Chief Information Officer Council Version 1.0, 2001.

[3] A. Tang, J. Han, P. Chen, "A Comparative Analysis of Architecture Frameworks", Centre for Component Software and Enterprise Systems, 2004.

[4] Allen Sayles, "Development of Federal Enterprise Architecture Framework using IBM Rational Unified Process and the Unified Modeling Language", Rational Brand Services, IBM Software Group, 2003.

[5] David C. Hay, "The Zachman Framework," Essential Strategies, Inc., 2000.

[6] Department of Defense, "DoD Architecture Framework Version 1.5: Volume I: Definitions and Guidelines," 2007.

[7] Department of Defense, "DoD Architecture Framework Version 1.5: Volume II: Product Descriptions," 2007.

[8] Department of Treasury, Chief Information Officer Council, Treasury Enterprise Architecture Framework, Version 1, 2001.

[9] Fatma Dandashi, Rolf Siegers, Judith Jones, Terry Blevins, "The Open Group Architecture Framework and the US Department of Defense Architecture Framework", Published by the Open Group, 2006.

[10] Federal Enterprise Architecture Program Management Office (2007). FEA Practice Guidance.

[11] Federal Enterprise Architecture, Chief Information Officer Council, Version 1.0, 2001.

[12] Frank Goethals, "An Overview of Enterprise Architecture Framework Deliverables", Information Systems Frontiers, Volume 8, Number 2, pp. 67-79(13), 2006.

[13] J. A. Zachman, “The Zachman Framework For Enterprise Architecture: Primer for Enterprise Engineering and Manufacturing”, 2003.

[14] J. A. Zachman, Viewing and Communicating Information Infrastructure: Enterprise Architecture (EA), 1987.

[15] J. Schekkerman, “Trends in Enterprise Architecture”, Institute for Enterprise Architecture, Report on the Third Measurements, 2005.

[16] John Wu, “Frameworks and Models: The Myth” EA Consultant, 2006. [Wu 06]

[17] John Zachman’s: Foundations for Enterprise Architecture, A two-day briefing for business and IT executives, 2008.

[18] Jon Siegel and the OMG Staff Strategy Group, “Developing in OMG’s Model-Driven Architecture”, Object Management Group White Paper, Revision 2.6, 2001.

[19] K. Langenberg, A. Wegmann, "Enterprise Architecture: What Aspects is Current Research Targeting?" Technical Report, EPFL, 2004.

[20] Lee Deidre, James Flyzik, Federal Enterprise Architecture Framework Version 1.1, 1999.

[21] L. Bass et al. “Software Architecture in Practice”, Addison Wesley, Boston, MA, USA, 2nd edition, 2003. [Bass et al. 03]

[22] L. Urbaczewski, "A comparison of Enterprise Architecture Frameworks", Issues in Information Systems, Vol. 7, No. 2, 2006.

[23] M. Ekstedt, P. Johnson, A. Lindstrom, M. Gammelgard, E. Johansson, L. Plazaola, E. Silva, J. Liliesköld, “Consistent Enterprise Software System Architecture for the CIO - A Utility-Cost Based Approach”, 2004.

[24] Oddmn Pauline Ohren, “An Ontological Approach to Characterizing Enterprise Architecture Frameworks”, ICEIMT, 2004.

[25] Osvalds, Gundars "Manuscript Instructions/Template for 2001", Proceedings of the 6th International Symposium, INCOSE 1996. ... [Osv01], 2001.

[26] Paula J. Hagan, "Guide to the (Evolving) Enterprise Architecture Body of Knowledge", MITRE Corporation. 2004.

[27] R. Sessions, "Comparison of the top Four Enterprise Architecture Methodologies" EA Comparison ObjectWatch, Inc., 2007.

[28] Richard V. McCarthy, “Toward a Unified Enterprise Architecture Framework: an Analytical Evaluation”, Issues in Information Systems, Vol. 7, No. 2, 2006.

[29] Rob Thomas, Raymond A. Beamer, Paula K. SowellCivilian Application of the DOD C4ISR Architecture Framework: A Treasury Department Case Study.

[30] Ruth Malan and Dana Bredemeyer, “Guiding Principles for Enterprise Architects” Architecture Discipline, 2004.

[31] S. Abdallah, G. H. Galal-Edeen, "Towards a Framework for Enterprise Architecture Frameworks Comparison and Selection", Fourth Int'l Conf. on Informatics and Systems, 2006.

[32] S. Leist, .G. Zellner, “Evaluation of Current Architecture Frameworks”, SAC’06, April, 23-27, Dijon, France, 2006.

[33] Schekkerman Jaap., "How to survive in the jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework", Third Edition, Trafford, 2006. [34] Schekkerman Jaap., "The Economic Benefits of Enterprise Architecture, How to quantify and Manage the economic Value of Enterprise Architecture". Trafford Publishing, Canada, 2005.

[34] The Open Group Architecture Framework Version 8, Enterprise Edition, 2002.

[35] The Open Group Architecture Framework Version 8.1.1, Enterprise Edition, 2006.