

Enterprise Application Architecture

The field of enterprise architecture essentially started in 1987, with the publication in the IBM Systems Journal of an article titled "A Framework for Information Systems Architecture," by J.A. Zachman. In that paper, Zachman laid out both the challenge and the vision of enterprise architectures that would guide the field for the next 20 years. The challenge was to manage the complexity of increasingly distributed systems. As Zachman said: "*The cost involved and the success of the business depending increasingly on its information systems require a disciplined approach to the management of those systems*". Zachman's vision was that business value and agility could best be realized by a holistic approach to systems architecture that explicitly looked at every important issue from every important perspective. His multiperspective approach to architecting systems is what Zachman originally described as an information systems architectural framework and soon renamed to be an enterprise-architecture framework.

The evolution of enterprise architecture happened because at that time system development had to address two core problems:

- System complexity—Organizations were spending more and more money building IT systems
- Poor business alignment—Organizations were finding it more and more difficult to keep those increasingly expensive IT systems aligned with business need.

Enterprise: An organizational unit – from a department to a whole corporation.

Architecture: A formal description of a system, or a detailed plan of the system at component level to guide its implementation. The structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time.

Enterprise Architecture: It is a formal description of an enterprise, a detailed map of the enterprise at component level to guide its changes. It provides details regarding the structure of an enterprise's components, their inter-relationships, and the principles and guidelines governing their design and evolution over time. Enterprise Architecture is about understanding all of the different components that go to make up the enterprise and how those components inter-relate.

Enterprise application (EA) (popularly known as enterprise software or enterprise application software (EAS)) is applications that a business uses to support the organization in order to solve enterprise problems. Normally, EA denotes a software platform that is quite complex and comparatively huge for small business use. However, they are designed to be user-friendly and easy to use. They are database-centric and demand high requirements in terms of security, user-access and maintenance.

It is a large software system platform intended to work in a corporate atmosphere like business or government. They are component-based, complex, scalable, distributed and mission critical. EA software comprises of a collection of programs with common business applications and organizational modeling. EAs are developed with enterprise architecture. The design of EA is accomplished in such a way that it can be successfully deployed across a

range of corporate networks (or intranets /Internet). It should assure to adhere with strict security requirements and administration management. Contrasting from the small business applications, which target particular business (e.g. billing software), an EA defines the business logic and supports the entire organization (including its departments), so as to lower costs and increase productivity as well as efficiency in the enterprise.

“At its heart, enterprise computing is all about combining separate applications, services and processes into a unified system that is greater than the sum of its parts.”

- Jim Farley et al. in *Java Enterprise in a Nutshell*

An Enterprise Application is a combination of other applications, services and processes that, when unified, form a application that is greater than the sum of its parts.”

- Paul Grenyer

Java is widely used for developing EA .

It is because it is named after(or illustrated by) coffee !!(Are you sure??)

Java has a vast array of excellent libraries for solving most of the common problems one needs to solve when developing enterprise applications. In many cases, there is more than one good choice for addressing a particular need, and oftentimes those libraries are free and open source under a business-friendly license.

Many companies choose Java for developing EA because:

Learning Curve

Java is a simple language with a huge support base. Enterprise projects tend to involve large numbers of developers and it is much easier to get a developer to a minimum level of competence in Java than other languages like C++. A lot of university graduates have been largely schooled in Java.

Choice & Reuse

Java has a vast array of libraries, frameworks, tools and IDEs, and server providers. To an enterprise it's good to have choice, even if that's just for use as a bargaining tool when negotiating price. The language lends itself to code quality tools that let implementation of corporate standards (and as mentioned there are a lot of those tools).

Platform Independence

Java is writing once, run everywhere. Sun has actively encouraged open standards that allow multiple vendors to implement their solutions. These standards give the customer the comfort that they can migrate from one vendor to another if a given vendor goes under or starts charging more. Of course the reality is that each vendor does their best to provide some "added value" features that tie the customer to them quite nicely.

Maturity

It has been around a long time, running a lot of servers. If your web application needs to be "6 sigma" or similar and you are the Mega Corp Chief Tech Officer, you are not going to look for developer wanting to do it in relatively immature platforms like RoR.

Timing/Marketing

Java was introduced to the world when programming was moving towards the web. It was positioned smartly and got a solid position early in web development. Because of the open standards, there are some very big companies producing these platforms and they market Java pretty hard to sell those platforms.

Inertia

Large corporations update/upgrade themselves very slow (many are still using Java 1.4), so once they have picked Java, it takes a huge investment to migrate to another platform. And convincing for change is to assure them to invest a lot of money for no immediate business benefit.

To recapitulate, Java ensures security, stability, robustness, scalability, platform independence, performance & resources usage, and Academic society wide acceptance.

Common Types of Enterprise Applications

Some of the more common enterprise applications include the following:

- Customer relationship management (CRM)

Wikipedia says - CRM is a model for managing a company's interactions with current and future customers. It involves using technology to organize, automate, and synchronize sales, marketing, customer service and technical support. It involves all kinds of interaction that a business corporation has with its client, whether it is sales or service-related. CRM enables the business by: Understand the customer, Retain customers through better customer experience, Attract new customer, Win new clients and contracts, and Increase profitably, Decrease customer management costs.

How CRM fits in with your Online Strategy



Fig: CRM <<http://crmconsulting.net.au>>

- Automated billing systems
- Enterprise Resource Planning
- HR Management

Enterprise Application Architecture (EAA)

Enterprise Architecture describes how organizational, information and technology structures support the strategy and operations of organizations. The EAA architecture of a business is a explanation of the structure of the application & software usage through the organization. The systems are decomposed into subsystems and connected/related to each other based on their interaction. The relationships with external environment, guidelines, users, terminology are also specified as a part of enterprise architecture of Software Systems. During the designing of EAA, an effective and thorough survey of existing systems should be performed by a team of high technical analysts in order to understand these systems and their interactions with other applications and databases.

Advantage/Need/Uses of enterprise application architecture

The purpose of enterprise architecture is to optimise across the enterprise the often fragmented legacy of processes (both manual and automated) into an integrated environment that is responsive to change and supportive of the delivery of the business strategy. Thus the primary reason for developing an EA is to get an overview (map) of the business' processes, systems, technology, structures and capabilities. We need an EA to provide a strategic context for the evolution of the IT system in response to the constantly changing needs of the business environment. We need an EA to achieve competitive advantage.

It acts for bridging the gap between Business and IT. It helps to enhance the relationships between IT and the business. It reinforces IT understanding of the business strategy. It creates a process for continuous IT/business alignment. Furthermore, it enhances IT agility to support business changes. And Hence, create business value from IT.

Describing and characterizing the architecture of an enterprise application is beneficial as it acts like a central document. This central document specifies an unambiguous understanding of the complete system and how it works as a whole. This aids in finding problems and solving them in order to speed up existing systems and enhance them in terms of performance. At time of introduction of a new application software or improvement of existing applications, the application architecture acts a key document for studying the effects of the changes in the entire enterprise. This document should be referred as a guide during the improvement/replacement of the current systems. Whenever an enterprise application is introduced into an existing system, the application architecture acts as the foundation for developing this software.

The value of EA:

- Business - IT and business – business alignment
- Change enabler
- Improved agility to enable a real time enterprise
- Standardisation, reuse and common principles, terms and work practices
- Integration and interoperability
- Structure, multiple perspectives and documentation

EA Bridges Strategy and Implementation

Architecture

- Business architecture
- Information architecture
- Solution architecture
- Technology architecture

Business Strategy

- Business drivers
- Business goals
- Business policy
- Trend analysis



Implementation

- Business processes
- Application systems
- Tech infrastructure
- Organizational structure

The bridge between strategy & implementation

Lecture 15 – Enterprise Architecture, TOGAF, Gartner,

TDT4252, Spring 2012



NTNU – Trondheim
Norwegian University of
Science and Technology

FEA
www.ntnu.edu

Enterprise Application vs. Desktop Application vs. Web Application

Desktop application runs on a single tier architecture. All the presentation logic, business logic and data storage and access logic resides in a single machine. Moreover, it can be a part of network. But the application and its database stay in the same machine. It does not support multiple users concurrently but only one user can access it at a time.

For beginners, web application and enterprise application seem to be similar although they are not. Enterprise application normally includes more than a single tier i.e. a multi-tier application. Multiple concurrent users can access the application.

For example Enterprise Application has at different tiers like Client, Presentation, Business Logic and Data tier. Client Tier can be Java clients, browser-based clients and mobile clients. Presentation Tier can be servlets, JavaBeans components and JSP components. Business Logic Tier can include servers, web services (SOAP, RESTful) ,etc. Data Tier can be of DBMS and LDAP.

Web application can be implemented without such complex enterprise. For example we can use various technologies to implement a web application such as Servlets, JSP, Hibernate, Maven, Databases, JavaBeans, etc. However, web applications suffer shortcomings when special kind of infrastructure services (functional or non-functional) are required as an essential part of the application. There are various applications with such special service requirements which require a business layer running over an infrastructure offering special services like: timer services, distributed transactions, remote method invocation, messaging

processing, etc. This kind of infrastructure services is not available in web servers. Hence, need arises for enterprise application that runs on a heavyweight application server (e.g. IBM WebSphere, Jboss and Geronimo).

Furthermore, Oracle website has put some clarifications regarding the standard edition of Java and Java Enterprise Edition. The major difference of Java EE from the standard edition is that Java EE requires an application server, instead of a web server like tomcat. The reason behind the need of application server is that, EJBs where the business logic resides needs a container. EJBs can be managed by application servers like Geronimo, IBM WebSphere and Jboss. These special application/containers can also handle web applications. Furthermore, an enterprise application holds enterprise beans (*an enterprise bean is a server-side component that encapsulates the business logic of an application*) and executes in a J2EE Container which caters security and transaction services to the beans. The associated file is an .ear file. Whereas a web application runs in a web-container like tomcat, contains servlets/JSPs. The associated file is named with an extension of .war.

Enterprise Architecture Frameworks(EAF)

Frameworks help people organize and assess completeness of integrated models of their enterprises. An Architectural Framework gives a skeletal structure that defines suggested architectural artifacts, describes how those artifacts are related to each other, and provides generic definitions for what those artifacts might look like. EAF says:

- How to create and use an enterprise architecture.
- Principles and practices for creating and using architectural description of the system.

Purpose of Framework

- Organize integrated models of an enterprise
- Assess completeness of the descriptive representation of an enterprise
- Understand an organization or a system
- Assist in identification and categorization
- Provide a communication mechanism
- Help manage complexity
- Identify the flow of money in the enterprise

Zachman Enterprise Architecture Framework

The **Zachman Framework** is an EAF which provides a formal and highly structured way of viewing and defining an enterprise. It consists of a two dimensional classification matrix based on the intersection of six communication questions (What, Where, When, Why, Who and How) with five levels of reification, successively transforming the most abstract ideas (on the Scope level) into more concrete ideas (at the Operations level). The Zachman Framework is a schema for organizing architectural artifacts (in other words, design documents, specifications, and models) that takes into account both whom the artifact targets (for example, business owner and builder) and what particular issue (for example, data and functionality) is being addressed. The Zachman Framework is not a methodology in that it does not imply any specific method or process for collecting, managing, or using the

information that it describes

	1. What (data)	2. How (function)	3. Where (network)	4. Who (people)	5. When (time)	6. Why (motivation)
1. Scope (context)						
2. Business model (concept)						
3. System model (logical)						
4. Technology model (physical)						
5. Detailed representation (component)						
6. Real system, i. e. executing the game of baseball						

Fig: An Empty Zachman Framework

The rows present:

- different perspectives of the enterprise
 - different views of the enterprise
 - different roles in the enterprise
1. Scope describes the system's vision, mission, boundaries, architecture and constraints. The scope states what the system is to do. It is called a black box model, because we see the inputs and outputs, but not the inner workings.
 2. Business model shows goals, strategies and processes that are used to support the mission of the organization.
 3. System model contains system requirements, objects, activities and functions that implement the business model. The system model states how the system is to perform its functions. It is called a white box model, because we see its inner workings.
 4. Technology model considers the constraints of humans, tools, technology and materials.
 5. Detailed representation presents individual, independent components that can be allocated to contractors for implementation.
 6. Real system depicts the operational system under consideration.

Columns present the various aspects of the enterprise.

1. What (data) describes the entities involved in each perspective of the enterprise. Examples include equipment, business objects and system data.
2. How (functions) shows the functions within each perspective.

3. Where (networks) shows locations and interconnections within the enterprise. This includes major business geographical locations, networks and the playing field.
4. Who (people) represents the people within the enterprise and metrics for assessing their capabilities and performance. The design of the enterprise organization has to do with the allocation of work and the structure of authority and responsibility.
5. When (time) represents time, or the event relationships that establish performance criteria. This is useful for designing schedules, the processing architecture, the control architecture and timing systems.
6. Why (motivation) describes the motivations of the enterprise. This reveals the enterprise goals, objectives, business plan, knowledge architecture, and reasons for thinking, doing things and making decisions.

Government Enterprise Architecture Frameworks (GEAFs)

In the context of government enterprises: a coordinated set of activity areas involving one or more public organizations and possibly third-party entities from private organizations or civil society, an EA provides technical descriptions of the organizational goals, business and administrative processes, information requirements, supporting applications and technology infrastructure of the enterprise. These descriptions are typically captured in the form of models, diagrams, narratives, etc. A Government Enterprise Architecture (GEA) may be associated with a single agency or span functional areas transcending several organizational boundaries, e.g. health care, financial management and social welfare.

Reasons for developing enterprise architectures in government include:

1. Understanding, clarifying and optimizing the inter-dependencies and relationships among business operations, the underlying IT infrastructure and applications that support these operations in government agencies and in the context of specific government enterprises
2. Establishing a basis for agencies to share information, knowledge and technology and other resources or jointly participate in the execution of business processes
3. Optimizing ICT investment and business cases across the whole of government by enabling the opportunities for collaboration and sharing of assets, thus reducing the tendency for duplicated and poorly integrated IT resources and capabilities

There is increasing awareness on the importance of EA as most of the leading countries in e-government have well established EA programs. Presently, there are EA maturity models with defined relations to well known e-Government Maturity stages. The increasing popularity of EA practices by governments in both developed and developing countries is indicated by the different global surveys on EA.

Despite the popularity of the EA practice in the private sector and increasingly in the government, the EA discipline is relatively new, lacking foundational theories and models and characterized by multiplicity of frameworks and reference models; even lacking an agreement on the definition and scope of the subject matter. From the earlier orientation of EA as a technological optimization or standardization concern, EA has gradually evolved to a management practice with stronger emphasis placed on the organizational-IT alignment.

However, there are mixed results in terms of outcomes from EA initiatives. In general, demonstrating concrete benefits from EA program has been challenging for many organizations. This difficulty is attributed to lack of metrics for EA initiatives. Notwithstanding,

a number of successful EA initiatives have been reported by some governments, particularly, in Canada and the US. Unfortunately, comparing and analyzing these EA initiatives and cases is difficult in the absence of assessment frameworks, techniques and tools.

1. Aim

The project aims to provide policy guidelines for the development of Government Enterprise Architecture Frameworks (GEAFs), establish concrete requirements for such a framework in Macao, and provide recommendations on how elements of a Macao GEAF (MGEAF) could be built from existing Government EA Frameworks, Reference Models, Methods, and Modeling Framework. The project will also provide an example of agency-specific EA based on the recommended Macao framework.

2. Objectives

1. Improving understanding and contributing to the body of knowledge of GEA through foundational research
2. Enhancing EA practice by providing policy guidelines and development of Government EA Frameworks based on results from (1) with the supporting toolkit
3. Understanding the factors that contribute to wide adoption of EA practice within a government
4. Building capacities of government agencies and their architects through development of courseware for educational and training purposes as well as the use of tools in (2)
5. Dissemination of project output (1 through 4) through various channels including publications (books, journals, conference papers and technical reports), schools and courses, seminars, workshops and projects.

References:

<http://www.isummation.com/>

<http://en.wikipedia.org>

<http://stackoverflow.com>

<http://www.oracle.com/>

<http://msdn.microsoft.com/en-us/library/bb466232.aspx>

<http://www.egov.iist.unu.edu/cegov/projects/Strategy-Government-Enterprise-Architecture>

Lecture Notes on Enterprise architecture , Norwegian University of Science and Technology