

INFORMATION SYSTEMS DEVELOPMENT PROCESS

DEFINING ENTERPRISE ARCHITECTURE

The corporate strategy provides well-defined objectives that form the foundation of organizational structure. The operation model represents how an enterprise operates across process, organization, technology domains in order to deliver value to stakeholders. Supply chain excellence depends on “alignment” of supply chain strategy with corporate strategy. Information systems play a critical role in aligning supply chain strategy corporate strategy. The alignment process is described with emerging field of study known as “Enterprise architecture”.

Enterprise architecture is the field of study (1980s) that provides structured approaches to manage increasing enterprise systems complexity and align information technology with ever changing business needs. The MIT Center for Information Systems Research (CISR) defines **enterprise architecture** as “*the organizing logic for business processes and IT infrastructure reflecting the integration and standardization requirements of the company's operating model. The operating model is the desired state of business process integration and business process standardization for delivering goods and services to customers*”. The operating models defined in such a way are useful to establish requirements for reusable core capabilities and to guide IT investment decisions governance. Figure 1 illustrates various IT characteristics of operating models that are relevant to various corporate structures and strategies⁶.

Business process integration	High	Coordination (Subsidiaries model) <ul style="list-style-type: none"> • Business unit control over business process design • Shared customer, supplier, product data • Consensus processes for designing IT infrastructure services; IT application decisions are made in business units 	Unification (Global standard quality) <ul style="list-style-type: none"> • High-level process owners design standardized process • Centrally mandated databases • IT decisions made centrally
	Low	Diversification (Conglomerates, holding company model) <ul style="list-style-type: none"> • Business unit control over business process design • Few data standards across business units • Most IT decisions made within business units. 	Replication (Franchisee model or Facility, process replication) <ul style="list-style-type: none"> • Centralized (or federal) control over business process design • Standardized data definitions but data locally owned with some aggregation at corporate • Centrally mandated IT services

Low Business process standardization High

Figure 1: IT characteristics of operating models

The operating model based on the corporate strategy provides necessary dialogue to evolve business architecture and IT architecture, the two major components of enterprise architecture. There are many methodologies ⁷ that are available to create enterprise architectures. The four important methodologies are:

- 1 *The Zachman Framework for Enterprise Architectures*: It provides *taxonomy* for developing enterprise architecture.
- 2 *The Open Group Architectural Framework (TOGAF)*: It provides *process* for developing enterprise architecture.
- 3 *The Federal Enterprise Architecture*: It is *implemented* enterprise architecture by US government agencies. Useful for government enterprises.
- 4 *The Gartner Methodology*: A practice model that can be adopted by business enterprises

⁶Jeanne W. Ross, Peter Weill and David C. Robertson, *Enterprise Architecture as Strategy* (Harvard Business School Press, 2006)

⁷Roger Sessions, *A Comparison of the Top Four Enterprise-Architecture Methodologies*, MSDN, 2007

The open group architecture framework (TOGAF) is useful in understanding the both the business and information technology dimensions of the information systems. According to TOGAF, *enterprise architecture* can be used to denote both an entire enterprise:

- encompassing all of its information and technology services, processes, and infrastructure
- and a specific domain within the enterprise

TOGAF relevance to Supply chain information systems⁸

In both cases, the architecture crosses multiple systems, and multiple functional groups within the enterprise. An extended enterprise nowadays frequently includes partners, suppliers, and customers. *If the goal is to integrate an extended enterprise, then the enterprise comprises the partners, suppliers, and customers, as well as internal business units.*

The business operating model concept is useful to determine the nature and scope of the enterprise architecture within an organization. Large corporations and government agencies may comprise multiple enterprises, and may develop and maintain a number of independent enterprise architectures to address each one. However, there is often much in common about the information systems in each enterprise, and there is usually great potential for gain in the use of a common architecture framework. For example, a common framework can provide a basis for the development of an Architecture Repository for the integration and re-use of models, designs, and baseline data.

There are four architecture domains that are commonly accepted as subsets of overall enterprise architecture, all of which TOGAF is designed to support:

- 1 The *Business Architecture* defines the business strategy, governance, organization, and key business processes.
- 2 The *Data Architecture* describes the structure of an organization's logical and physical data assets and data management resources.
- 3 The *Application Architecture* provides a blueprint for the individual applications to be deployed, their interactions, and their relationships to the core business processes of the organization.
- 4 The *Technology Architecture* describes the logical software and hardware capabilities that are required to support the deployment of business, data, and application services. This includes IT infrastructure, middleware, networks, communications, processing, standards, etc.

The TOGAF Architecture Development Method (ADM) provides a tested and repeatable process for developing architectures. The ADM includes establishing an architecture framework, developing architecture content, transitioning, and governing the realization of architectures.

⁸ <http://pubs.opengroup.org/architecture/togaf9-doc/arch/>

All of these activities are carried out within an iterative cycle of continuous architecture definition and realization that allows organizations to transform their enterprises in a controlled manner in response to business goals and opportunities.

5.1 CHOOSING APPROPRIATE SYSTEM DEVELOPMENT METHODOLOGIES

Different stakeholders work together to create a solution architecture. The complex solution architecture requires additional structure or methodology for design and development called as *systems development life cycle*. There are three popular methodologies - *waterfall*, *agile* and *rational unified process*. These methodologies form the framework for planning and controlling the creation of an information system. All the three methodologies are structured around the following generic process activities as listed in figure 2:

1. Business requirements gathering
2. Software specification description based on requirements
3. Developing system architecture to meet requirements
4. Implementation (coding)
5. Testing
6. Documentation
7. Software training and support
8. Maintenance

Waterfall methodology: In this methodology the generic process activities as listed in figure 2 are performed sequentially. It is the conventional systems development model. The model fails if the design flaw is not discovered before deployment. This causes huge project delays, overruns, and even the risk of project failures. To overcome this drawback, the waterfall model is replaced with an iterative model. In the iterative model, at each step the requirements are verified and if flaws are discovered, the step is repeated until it meets the requirement.

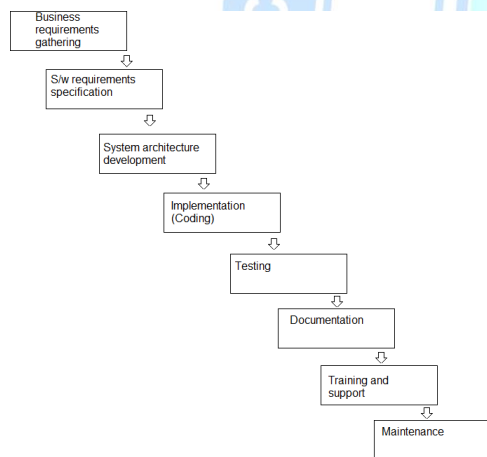


Figure 3: Waterfall model

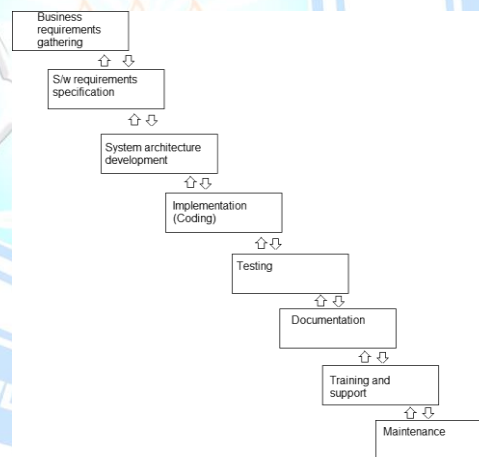


Figure 4: Iterative model

This model requires direct communication with business stakeholders in the requirements gathering phase and there after relies on volumes of documentation for communication.

Agile methodology: This model tries to reduce the risk involved in waterfall model by developing software in short iterations with distinct deliverables. Each deliverable may take from one to four weeks. It follows a mini waterfall model with iterations and many times involves development of new functionality for each release of software application.

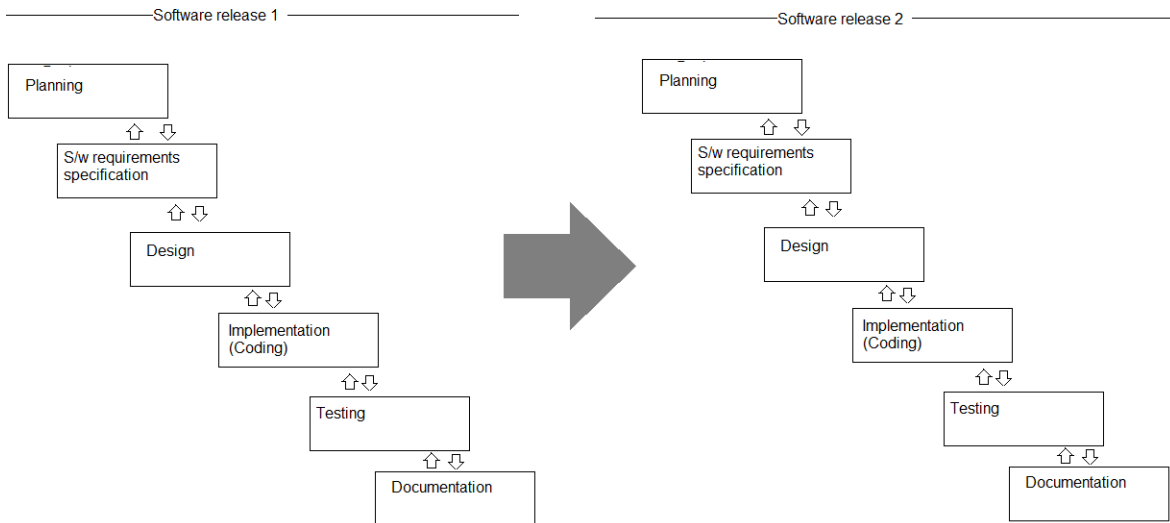


Figure 5: Agile methodology

This model emphasizes on direct communication with business stakeholders in real time instead of written documents. Most agile teams are co-located during the entire software development life cycle.

Rational unified process (RUP): This is a software product driven methodology (original Rational and now owned by IBM) developed by which can be customized by the enterprise based on its project needs. It is designed to build software in a succession of incremental iterations. The RUP has two dimensions:

1. *Workflows:* The horizontal dimension represents dynamic aspect of the process expressed in terms of cycles, phases, iterations, and milestones.
2. *Phases:* The vertical dimension represents static aspect of the process described in terms of process components: activities, disciplines, artifacts, and roles.

Figure 6 illustrates the RUP process with workflows and phases. The iterations in each of the phases have tasks that are categorized into the nine workflows. It captures many of modern software development's best practices as described below:

1. Develop software iteratively
2. Manage requirements
3. Use component-based architectures
4. Visually model software
5. Continuously verify software quality
6. Control changes to software

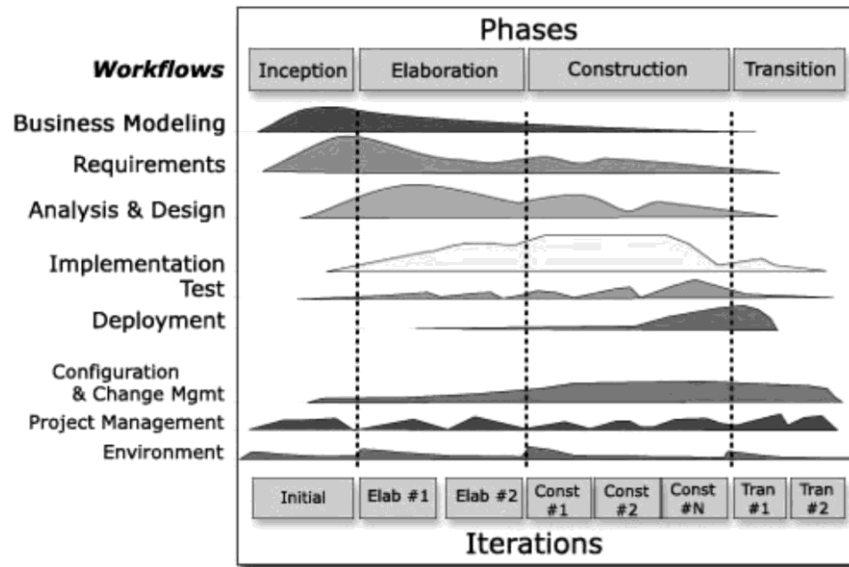


Figure 6: RUP software development process framework⁹

<http://www-128.ibm.com/developerworks/rational/library/content/RationalEdge/jan01/WhatIsTheRationalUnifiedProcessJan01.pdf>

5.2 ADOPTING RELEVANT SYSTEMS DEVELOPMENT MODEL

While the system development methodologies span full development life cycle, the development models provide important tactics for improving the quality of information systems. They are useful in designing information system that meets business requirements. There are three important development models:

1. Test driven development (TDD)
2. Model driven development (MDD)
3. Design patterns

Test driven development (TDD): This model is a result from agile methodology. The iterative process in TDD involves writing the test cases first and then implementing the code that is necessary to pass the test case. The test cases can be produced by customers as well as developers (systems engineers).

Model driven development (MDD): It is an approach where the code is generated for the primary software artifacts called “models” that describes a system from a particular perspective (textually or visually). These models are expressed using notations such as *unified modeling language (UML)* and *business process management notation (BPMN)*. Models expressed using BPMN are helpful to represent business processes and UML are helpful to represent technical details.

Design patterns: This model is helpful for systems engineers in overcoming design phase problems. Design patterns reduce code and complexity, and can facilitate communication during

design. They are reusable templates or descriptions that can be used in many situations to solve problems in software design context. In general these patterns are defined specific to a domain and are shared by design and development teams. The repositories of design patterns include patterns with the following elements: *name, context, problem, solution and variations, participants and collaborators, consequences, implementation and generic structure.*

Enterprises select appropriate methodology for enterprise architecture, right development methodology and suitable development model based on their needs. Expert support is sought in taking such decisions. A well balanced combination of enterprise architecture, development methodology and development model will not only increase the speed of the information system project but also improves its quality and reduces development risk.

5.3 SUMMARY

There are multiple stakeholders from business and information technology perspectives in enterprise information systems. Research indicates that as the complexity of information systems increase, enterprises have to follow structured methodologies to align information systems objectives with business objectives. The information systems development process can be viewed from three parts: Development of enterprise architecture in alignment with business architecture, systems development methodologies and systems development models:

Enterprise architectures	Systems development methodologies	Systems development models
Zachman's framework	Waterfall	Test driven development
TOGAF	Agile	Model driven development
FEA	Rational unified process (RUP)	Design patterns
Gartner		

Enterprises choose appropriate enterprise architecture framework, suitable development methodology and right development model based on the information systems complexity, value to business, available time and resources.

5.4 SELF-ASSESSMENT QUESTIONS

1. How is business operating model related to enterprise architecture?
2. Discuss about the four important enterprise architecture frameworks.
3. Discuss about three important systems development methodologies.
4. Write a brief note on three systems development models

