Enterprise Architektra

Informační management VIKMA07

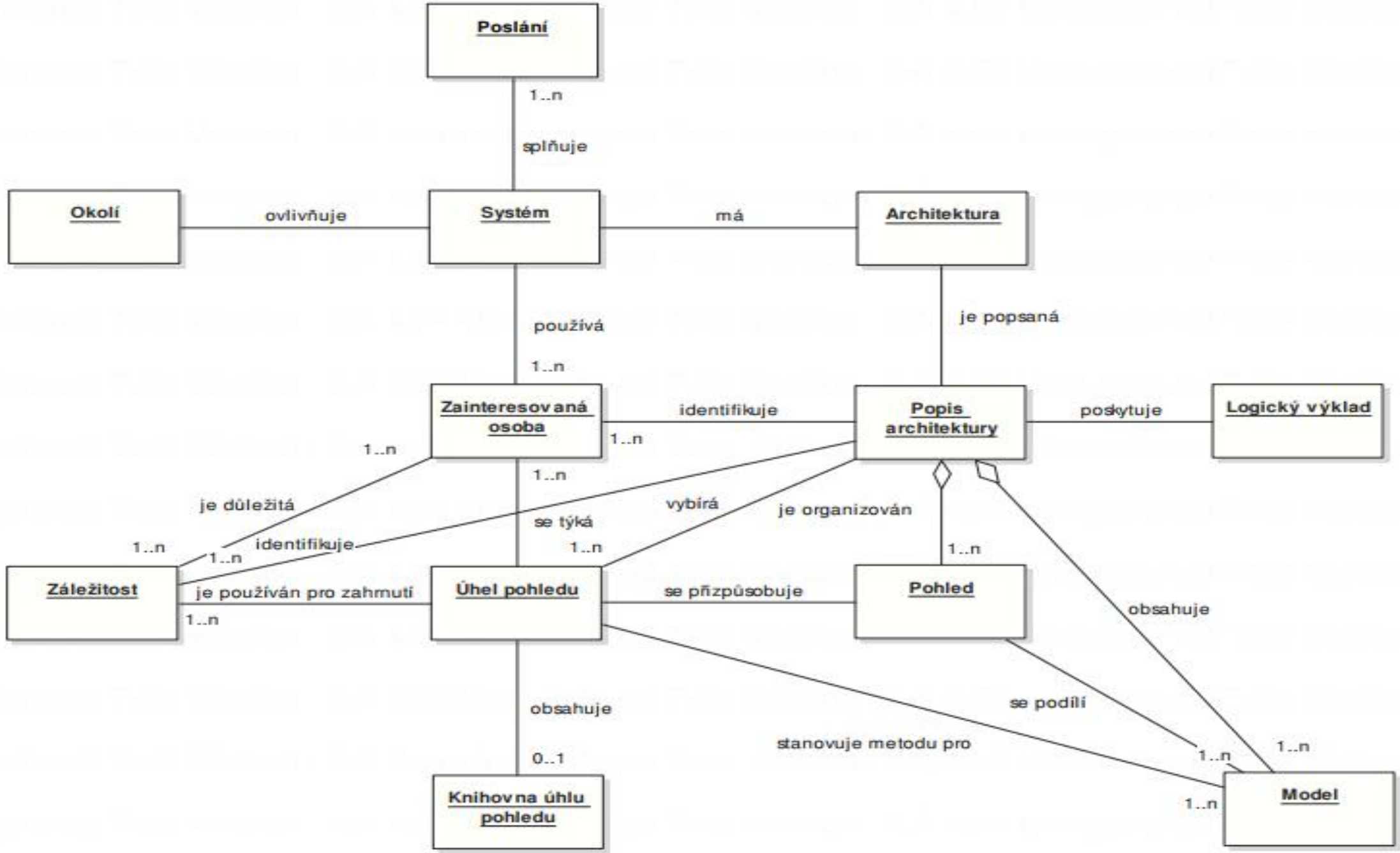
Mgr. Jan Matula, PhD.

177798@mail.muni.cz

III. blok

Architektura

- Formální popis systému nebo jeho detailní plán na úrovni komponent vedoucí k jeho implementaci.
- Struktura komponent a jejich vazeb včetně principů a návodů, které řídí návrh a rozvoj v čase.
- Definice podnikové architektury (Enterprise Architecture) vychází z normy ISO 42010, která uvádí, že architekturou se myslí: „základní organizace systému ztělesněná jeho složkami, jejich vzájemnými vztahy a vztahy k prostředí a principy, kterými se řídí její návrh a vývoj.


Poslání
1..n
splňuje
Okolí
ovlivňuje
Systém
má
Architektura
používá
1..n
je popsaná
Zainteresovaná osoba
identifikuje
1..n
Popis architektury
poskytuje
Logický výklad
1..n
1..n
je důležitá
1..n
identifikuje
1..n
se týká
vybírá
je organizován
1..n
Záležitost
1..n
je používán pro zahmutí
1..n
Úhel pohledu
se přizpůsobuje
Pohled
obsahuje
obsahuje
0..1
se podílí
stanovuje metodu pro
1..n
1..n
1..n
Knihovna úhlu pohledu
Model

6 praktik pro popis architektury

- vztahovat dokumentaci k informaci o verzi,
- identifikovat osoby zainteresované na systému,
- vybrat správné architektonické pohledy,
- spojit je s relevantními úhly pohledu,
- dodržovat konzistenci mezi pohledy architektury,
- rozumně a promyšleně vybírat aktuální architekturu z daných alternativ.

Cíle EA

- Hlavním cílem podnikové architektury (EA) je zohlednit popis cílů organizace (definovaných v podnikové strategii), způsobů jak jsou tyto cíle dosahovány pomocí podnikových procesů a způsobů, jak mohou tyto procesy být podpořeny technologiemi.
- Vhodné nastavení podnikové architektury (resp. IT architektury) je základním předpokladem pro efektivní fungování celé organizace. Pokud není architektura nastavená správně, dochází k výraznému omezení rozvoje organizace, které může vést k omezení obchodní aktivity (což vede ke nížení příjmů nebo ke zvýšení nákladů na zajištění klíčových procesů).

Governance architektury

- Způsob jakým jsou podnikové architektury a další architektury řízeny a kontrolovány na úrovni podniku.

Spaghetti Architecture

The business benefits of Enterprise Architecture

A more efficient business operation

- Lower business operation costs
- More agile organization
- Business capabilities shared across the organization
- Lower change management costs
- Improved business productivity

A more efficient IT operation

- Lower software development, support, and maintenance costs
- Increased portability of applications
- Improved interoperability and easier system and network management
- Improved ability to address critical enterprise--wide issues like security
- Easier upgrade and exchange of system components

Better return on existing investment and reduced risk for future investment

- Reduced complexity in the business and IT
- Maximum return on investment in existing business and IT infrastructure
- Reduced risk overall in new investments and their cost of ownership

Vývoj konceptu podnikové architektury

- Kořeny podnikové architektury vychází z teorie IT sladění (IT alignment). Tento přístup reprezentuje neustálou cestu transformace podniku za dosažením strategických byznys cílů, které jsou prostřednictvím informačních technologií umožňovány, podporovány anebo dokonce stimulovány.

Evoluční stupně EA 1/3

- **Podniková architektura navrhuje organizaci IT** - Podniková architektura se v tomto evolučním stupni orientuje na návrh organizace informačního systému (IS). Předmětem zájmu architektonického návrhu jsou zdroje a schopnosti podnikové informatiky, tj. data, nástroje (hardware), metody a lidé (IT personál).

Evoluční stupně EA 2/3

- **Podniková architektura navrhuje organizaci podniku** - Podniková architektura v tomto evolučním stupni vystupuje z IT domény a orientuje se zároveň i na návrh organizace, tj. organizační infrastrukturu a procesy.
- Předmětem zájmu architektonického návrhu, vedle již zmíněných zdrojů a schopností podnikové informatiky, které byly architektonicky navrhovány v předchozím evolučním stupni a nyní by již v podniku měly být tyto činnosti dobře zvládnuty, jsou následující objekty: aktér; byznys pravidlo; byznys schopnost; čas; hybná síla; lokalita; majetek; metrika; organizační jednotka; politika a regulace; procesy; produkt; projekt; role řízení kvality, shody a udržitelnosti; taktika, úkol, znalost a informace.

Evoluční stupně EA 3/3

- **Podniková architektura podporuje návrh strategie podniku** - Podniková architektura se v tomto evolučním stupni podílí na formování strategie podniku a stává se prostředkem organizačních inovací a zajištění udržitelného rozvoje podniku. Předmětem jejího zájmu je návrh souladu strategie s charakteristikami okolí podniku, ať je již vnímáno jako množina prvků (zákazníci, dodavatelé, konkurence) nebo jako klíčové politické, legislativní, ekonomické, sociokulturní, technologické a environmentální faktory.
- Přitom se snaží architektonicky formovat nejen strategii samotnou, ale prostřednictvím inovací navrhovat změny právě i prvků a faktorů, které utvářejí okolí podniku. Ke klíčovým objektům, které jsou architektonicky navrhovány, patří vize a cíle podniku, strategie dosažení těchto cílů, ale také tzv. motivátory (externí omezení a podmínky, které motivují organizaci k formulaci cílů) a v neposlední řadě externí zainteresované strany (Stakeholders).

WHAT IS TOGAF?

- TOGAF is an open architecture Framework
  - Methods – ADM
  - Tools
- TOGAF documentation can be downloaded for free at www.opengroup.org

Architecture Framework

Rámec architektury

- Konceptuální struktura používající se pro rozvoj, implementaci a údržbu architektury.

GENERAL DEFINITION

- TOGAF is an architecture framework.
- Provides the methods and tools for assisting in the acceptance, production, use and maintenance of an enterprise architecture
- TOGAF is based on:
  - An iterative process model
  - A re-usable set of existing architecture assets
  - Supported by architectural best practices
- Architecture is:
  - The structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time

The advantages of using TOGAF

- TOGAF represents best practice in architecture development.
  - It has been developed through the collaborative efforts of 300 Architecture Forum member companies from some of the world's leading IT customers and vendors
- Using TOGAF as the architecture framework will allow architectures to be developed that are:
  - Consistent documentation
  - Reflect the needs of stakeholders
  - Employ best practice
  - Follow Current requirements and likely future needs of the business
- TOGAF helps to ‘‘demystify’’ and de-risk the architecture development process
- Provides a platform for adding value, and enables users to build genuinely open systems-based solutions to address their business issues and needs

Architecture domains

- Business Architecture
  - Defines the business strategy, governance, organization, and key business processes
- Data Architecture
  - Describes the structure of an organization’s logical and physical data assets and data management resources
- Application Architecture
  - A blueprint for the individual applications to be deployed, their interactions, and their relationships to the core business processes of the organization
- Technology Architecture
  - Describes the logical software and hardware capabilities that are required to support the deployment of business, data, and application services
  - Includes IT infrastructure, middleware, networks, communications, processing, standards, etc.

What does TOGAF Contain?

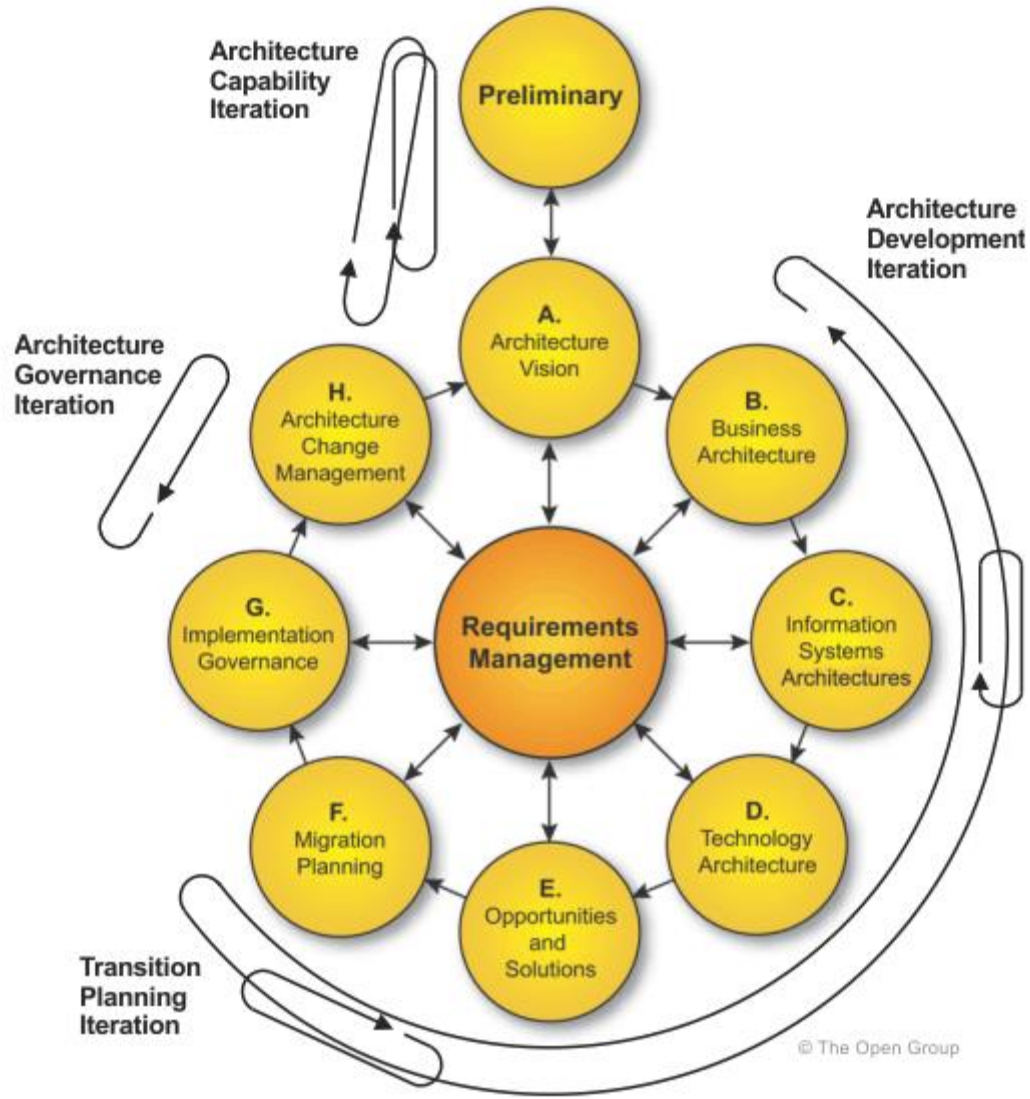
- TOGAF reflects the structure and content of an architecture capability within an enterprise.
- Definition of "**Capability**" - An ability that an organization, person, or system possesses. Capabilities are typically expressed in general and high-level terms and typically require a combination of organization, people, processes, and technology to achieve. For example, marketing, customer contact, or outbound telemarketing.

The Architecture Development Method (ADM)

- The ADM describes a process for deriving an organization-specific enterprise architecture that addresses business requirements.
- The ADM is the major component of TOGAF and provides guidance for architects on a number of levels:
- It provides a number of **architecture development phases** (Business Architecture, Information Systems Architectures, Technology Architecture) in a cycle, as an overall process template for architecture development activity.

The Architecture Development Method (ADM)

- It provides a narrative of each architecture phase, describing the phase in terms of objectives, approach, inputs, steps, and outputs. The inputs and outputs sections provide a definition of the architecture content structure and deliverables (a detailed description of the phase inputs and phase outputs is given in the Architecture Content Framework).
- It provides cross-phase summaries that cover requirements management.

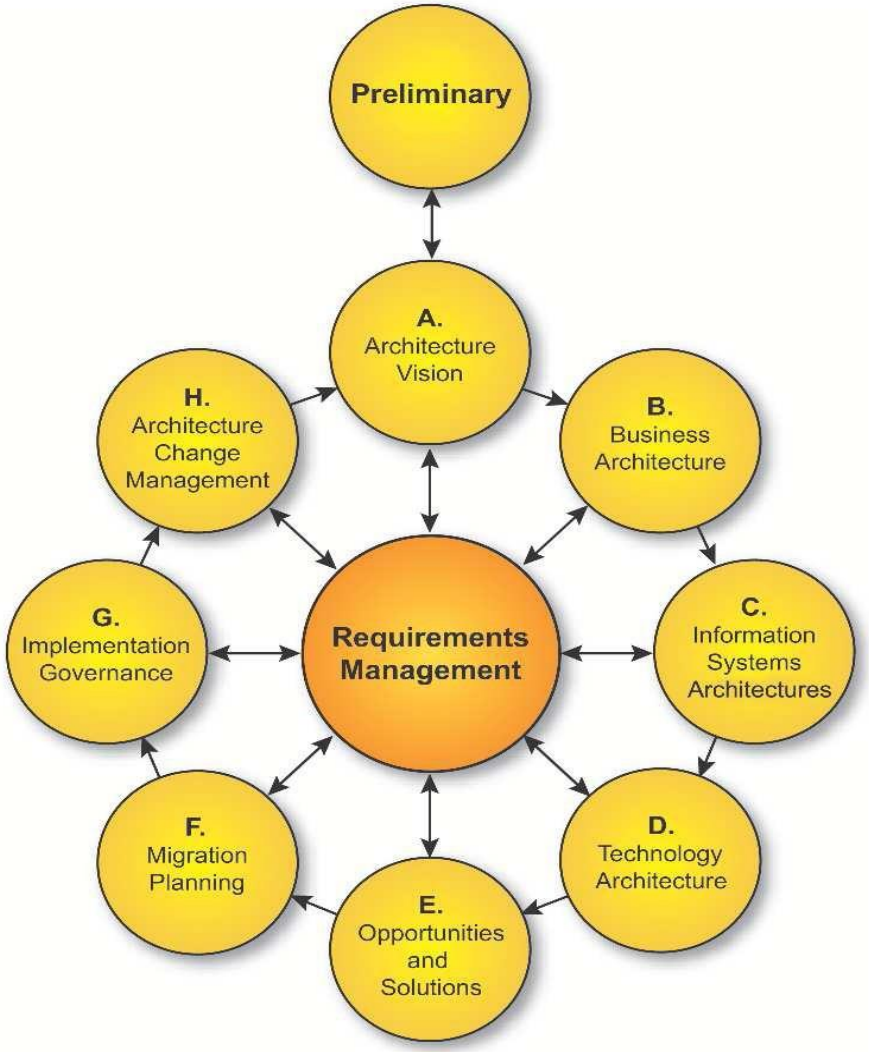

Architecture Capability Iteration
Preliminary
Architecture Development Iteration
A. Architecture Vision
Architecture Governance Iteration
H. Architecture Change Management
B. Business Architecture
G. Implementation Governance
Requirements Management
C. Information Systems Architectures
F. Migration Planning
D. Technology Architecture
E. Opportunities and Solutions
Transition Planning Iteration
© The Open Group

TOGAF ADM Structure

- A method for developing Enterprise Architecture is the ADM

ADM = Architecture Development Method

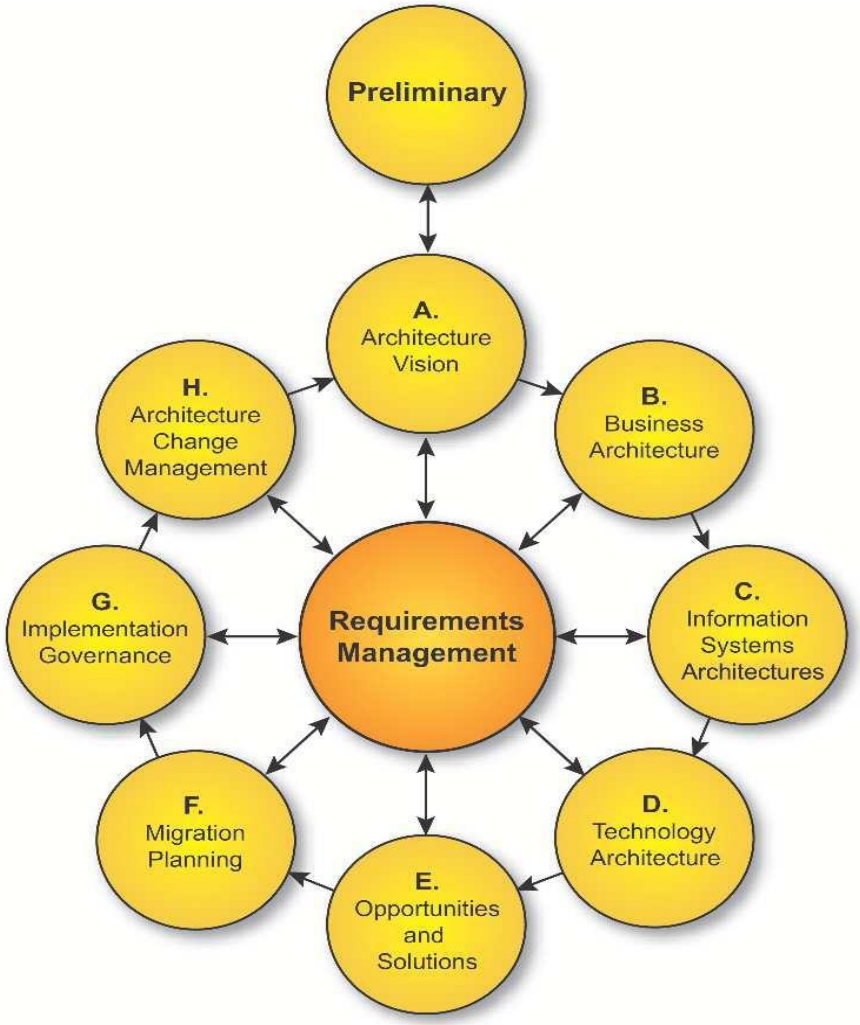
- The core of TOGAF
- A step-by-step approach to develop and use an enterprise architecture



TOGAF ADM Structure

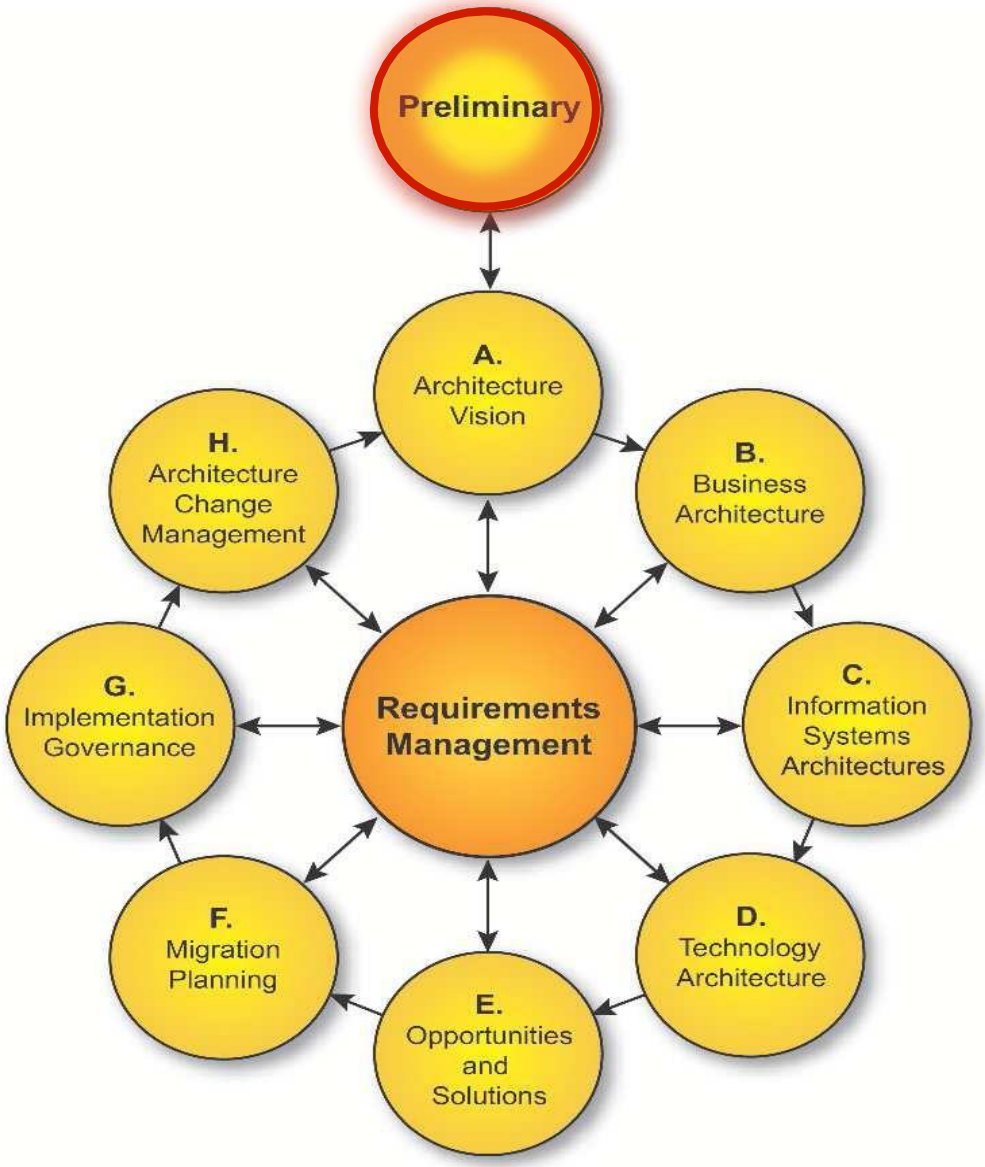
The ADM consists of 10 phases:

- The Preliminary Phase
- Phase A: Architecture Vision
- Phase B: Business Architecture
- Phase C: Information Systems Architectures
- Phase D: Technology Architecture
- Phase E: Opportunities & Solutions
- Phase F: Migration Planning
- Phase G: Implementation Governance
- Phase H: Architecture Change Management
- Requirements Management



ADM Preliminary phase

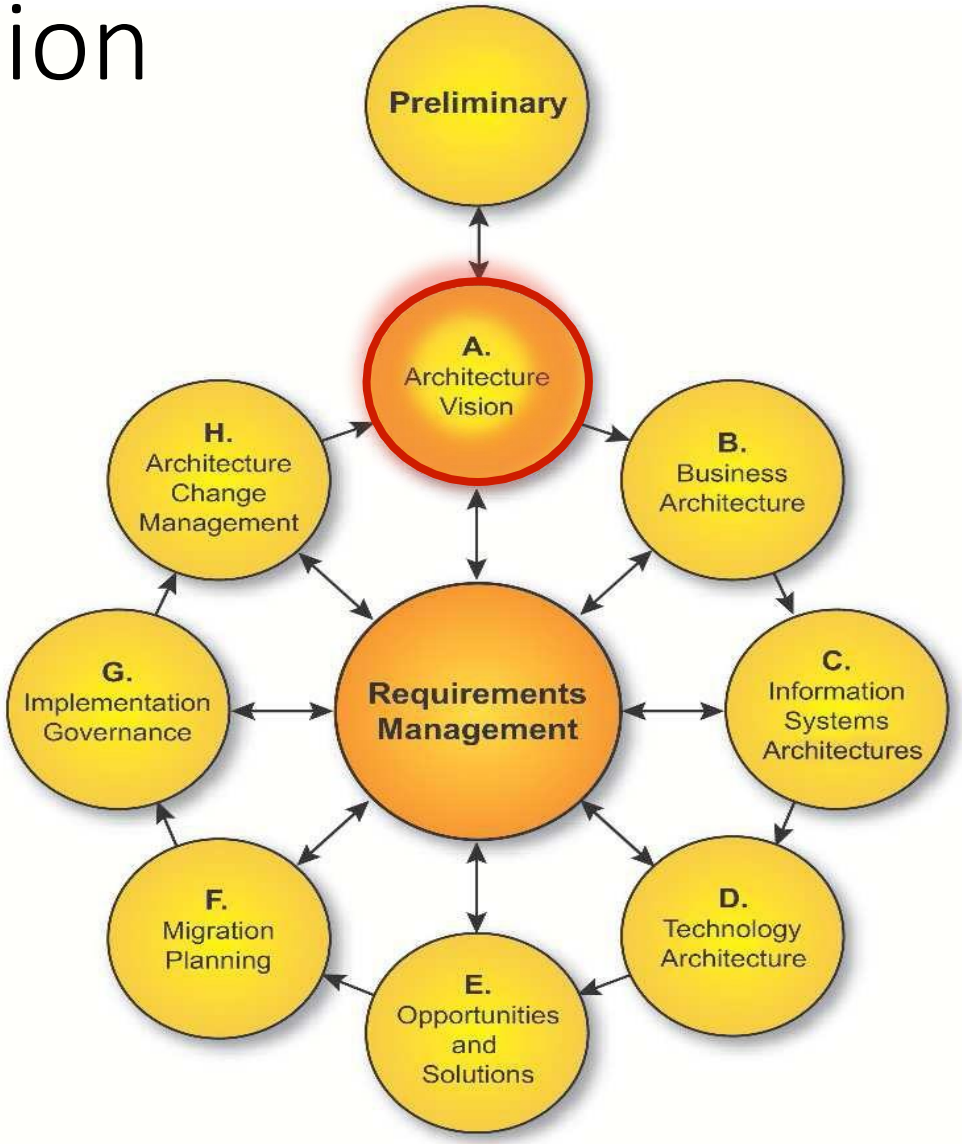
Describes the **preparation and initiation activities** required to create an **Architecture Capability** including customization of TOGAF and definition of **architecture principles**.



Phase A: Architecture Vision

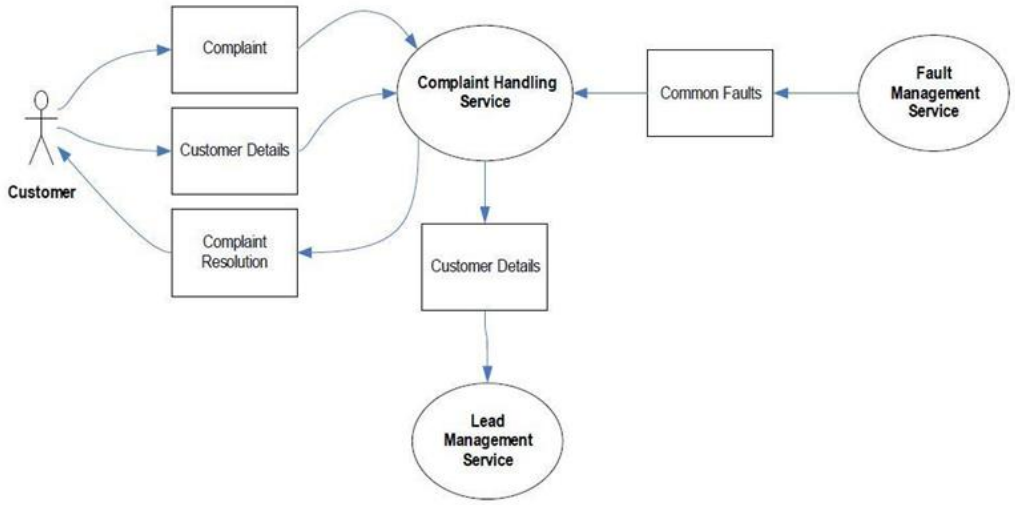
Phase A is the initial phase of an architecture development cycle

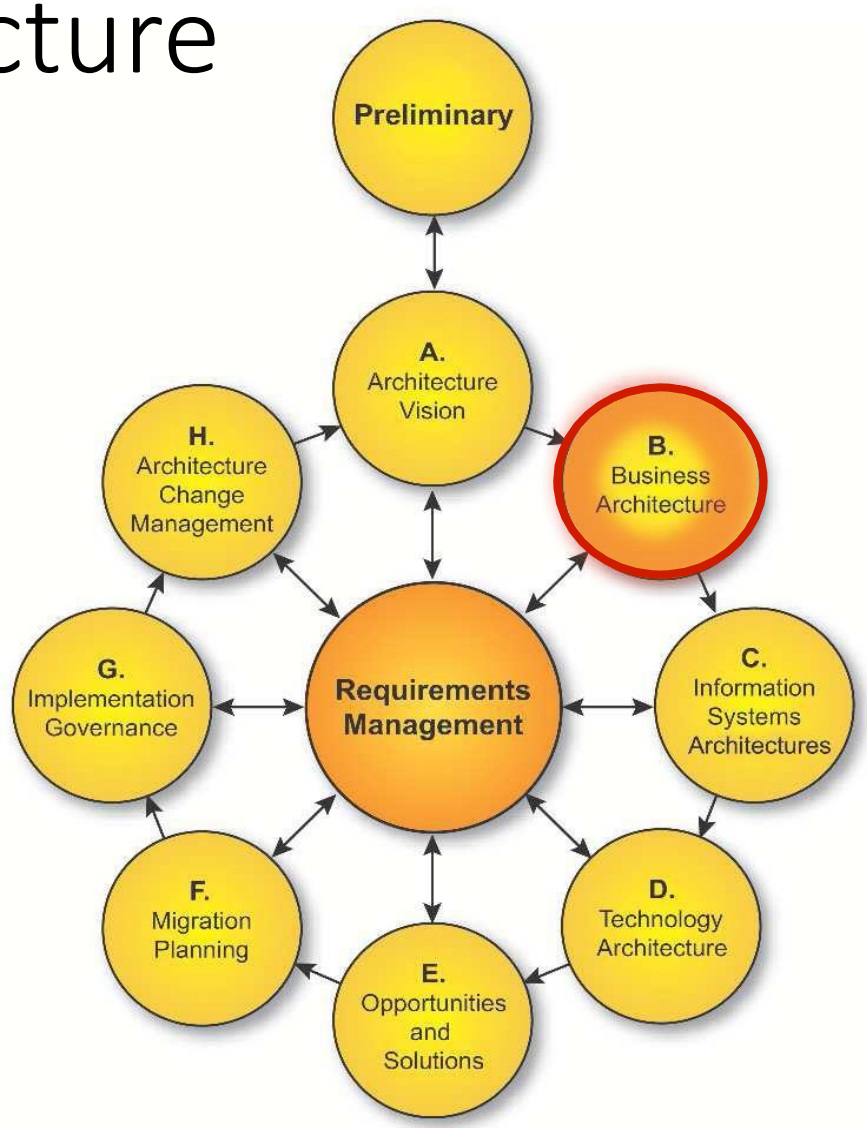
- Defining the scope of the architecture development initiative
- Identifying the stakeholders
- Creating the **Architecture Vision in accordance with business needs**
- **Obtaining approval** to proceed with the architecture development



Phase B: Business Architecture

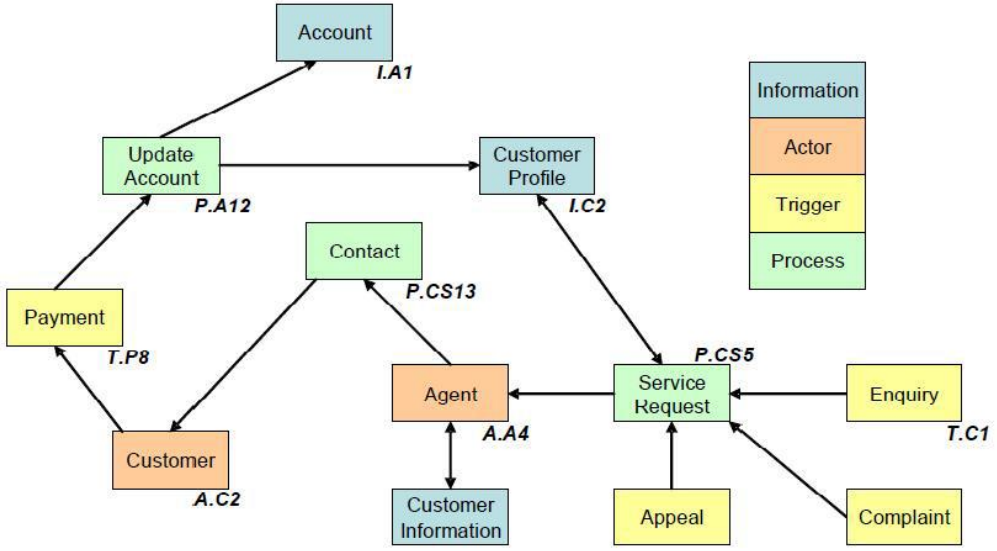
Phase B: Business Architecture describes the development of a **Business Architecture to support the agreed Architecture Vision**

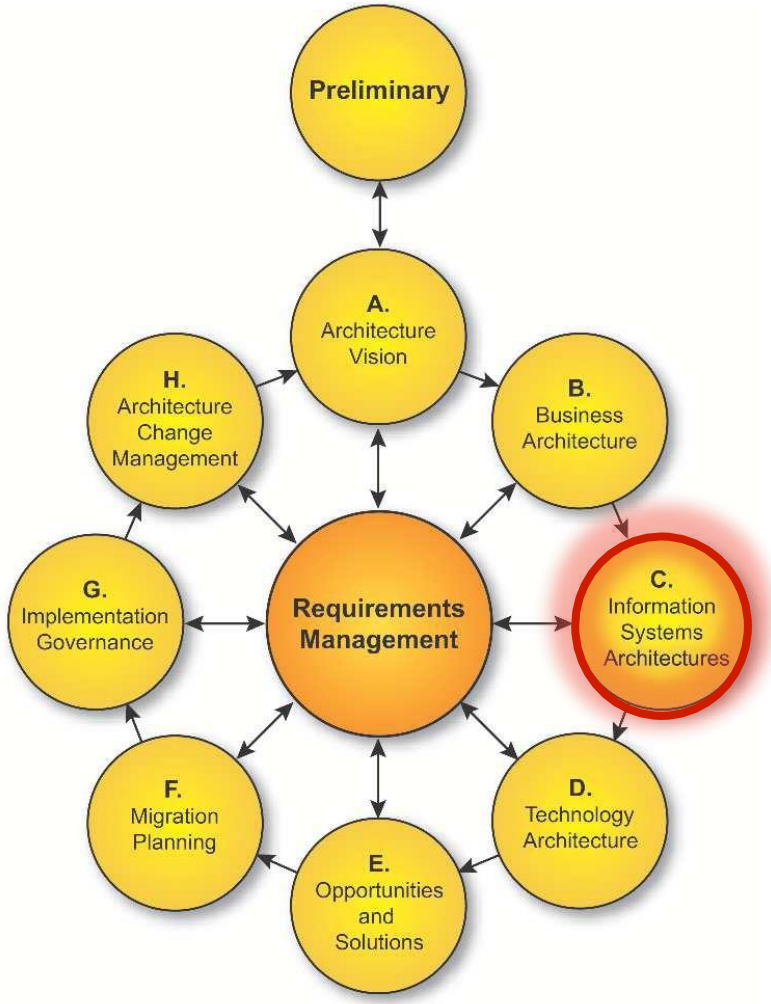




Phase C: Information Systems Architectures

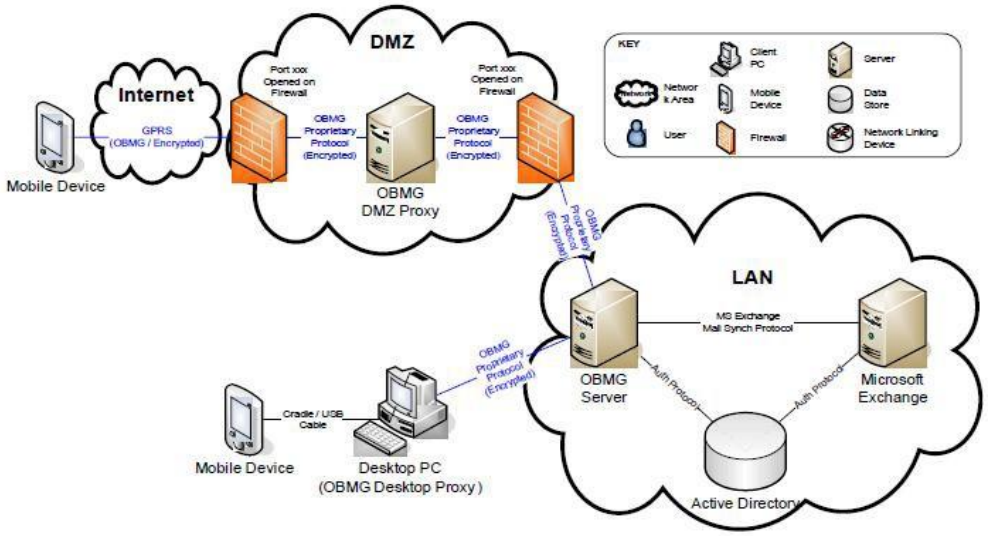
Phase C: Information Systems Architectures describes the development of **Information Systems Architectures to support the agreed Architecture Vision**

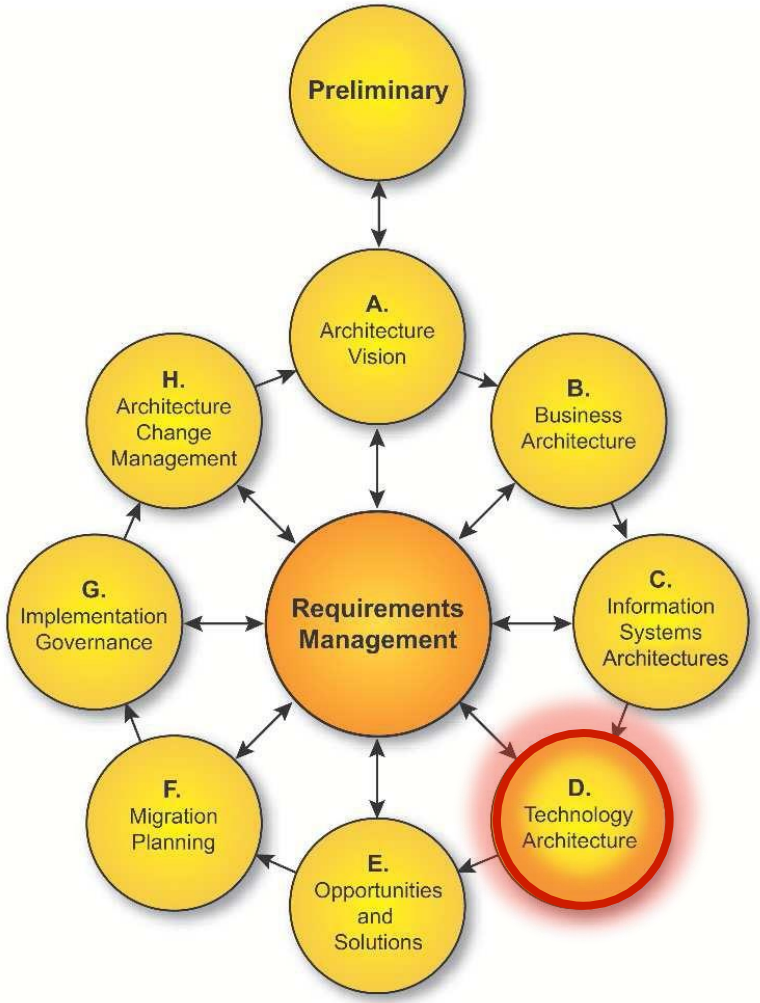




Phase D: Technology Architecture

Phase D: Technology Architecture describes the development of the **Technology Architecture to support the agreed Architecture Vision**

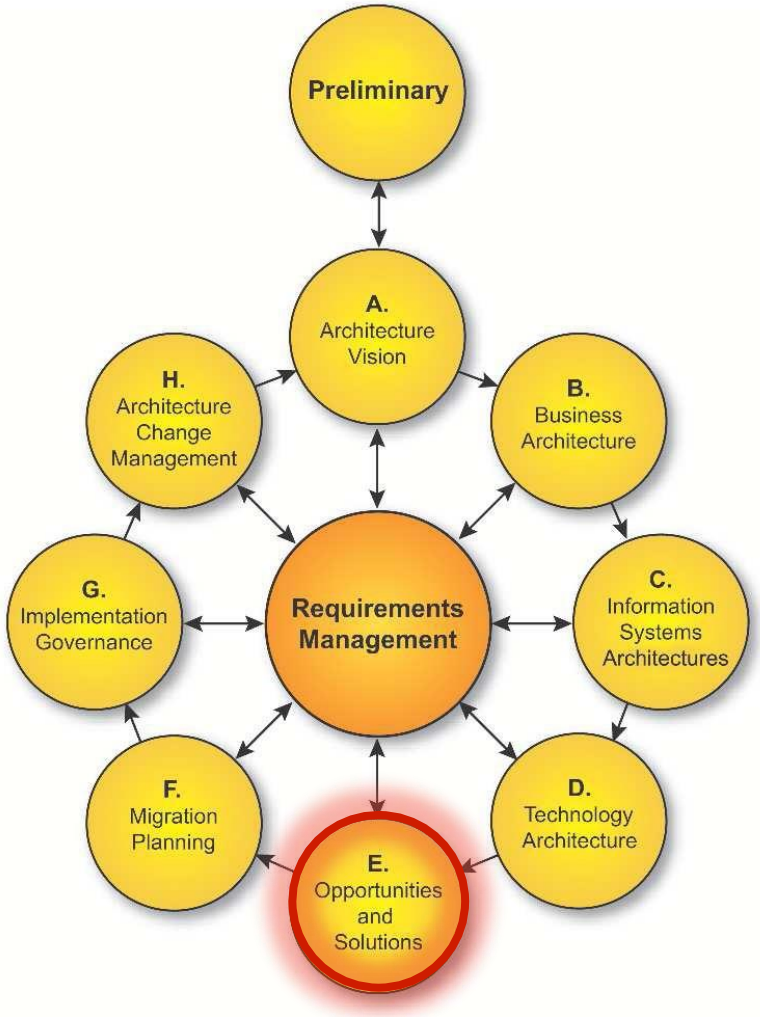




Phase E: Opportunities & Solutions

Phase E describes **initial implementation & migration planning**

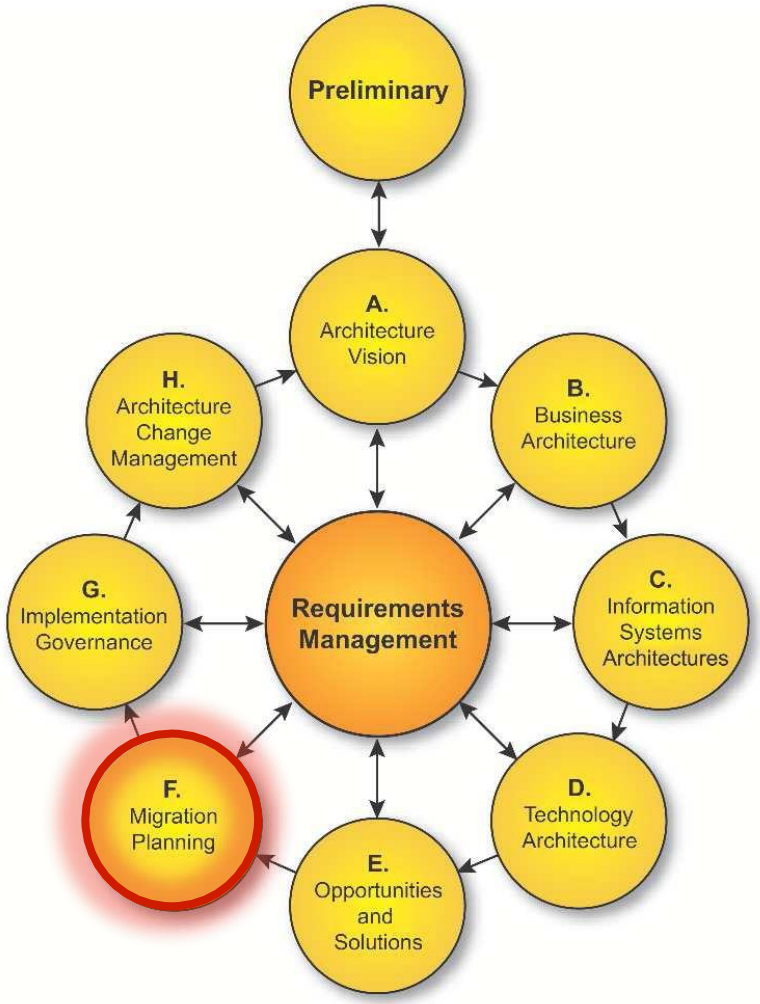
Includes the **identification of delivery packages** for the architecture defined in the previous phases



Phase F: Migration Planning

Phase F addresses **how to move from the Baseline to the Target Architectures by finalizing a detailed Implementation and Migration Plan**

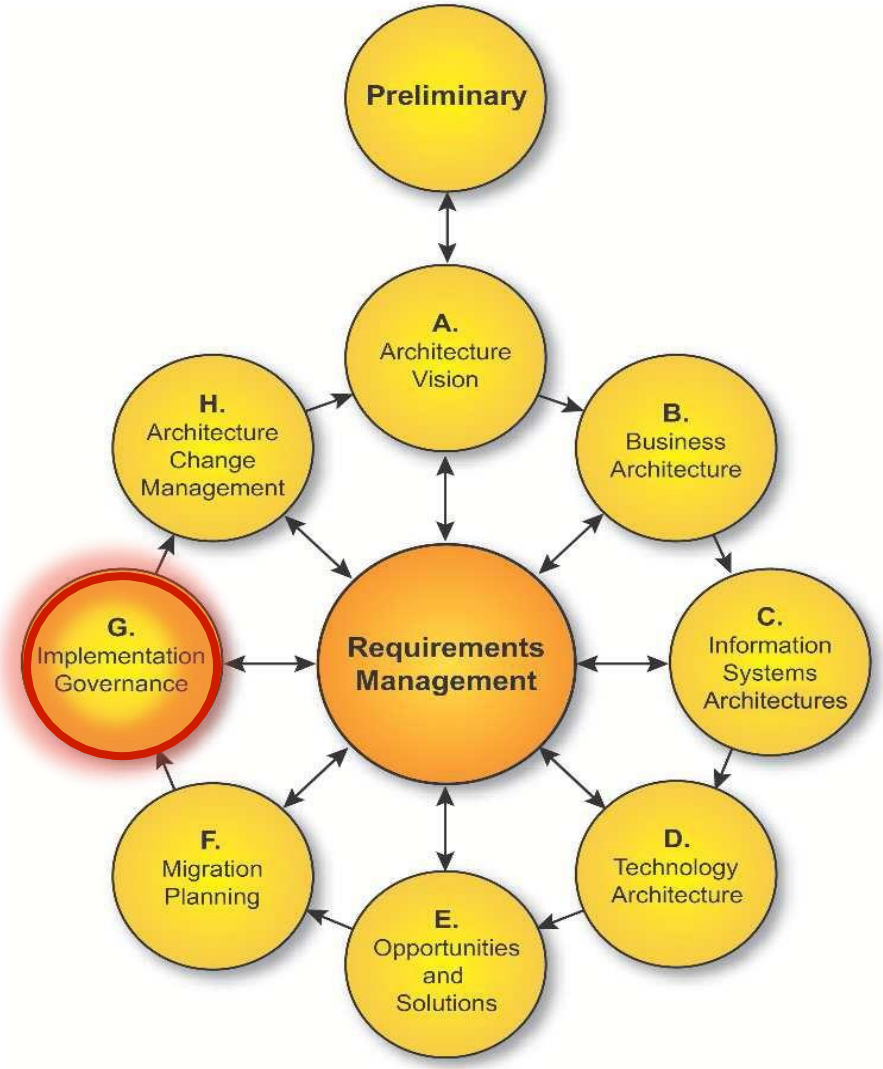
- PRIORITIZATION by cost / benefit



Phase G: Implementation Governance

Phase G: provides an architectural oversight of the implementation

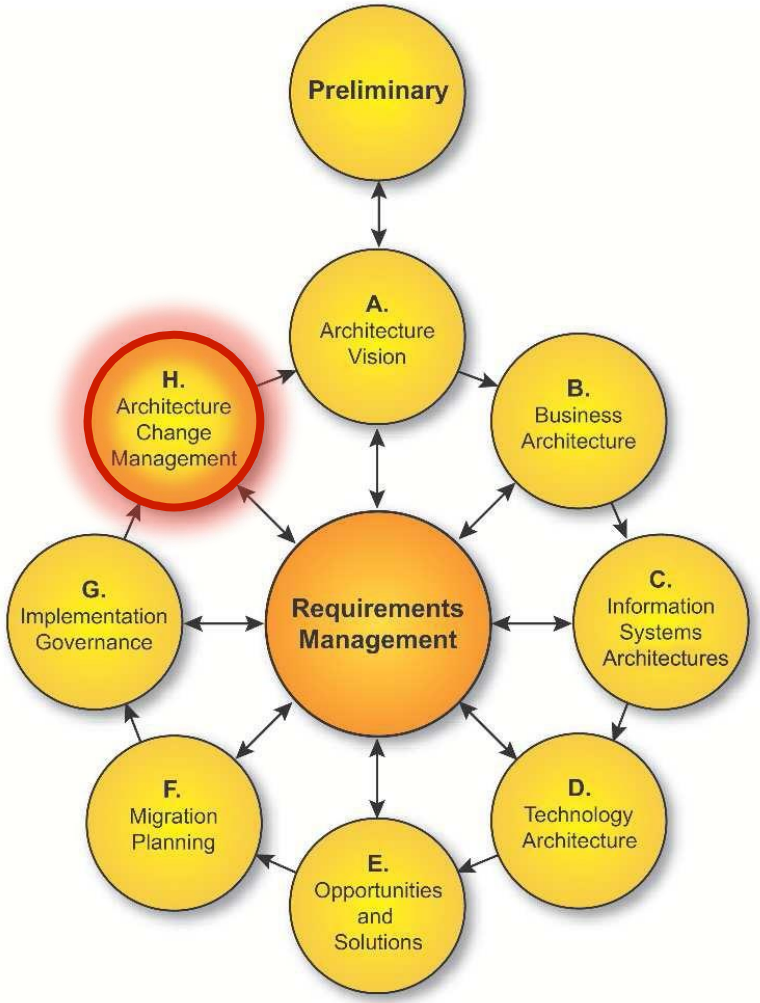
- CONTRACTS
- Compliance Review
- Handover for Solution Architects
- Project Recommendation
- Post-implementation review
- Risk management



Phase H: Architecture Change

Phase H: Architecture Change Management establishes procedures for managing change to the new architecture

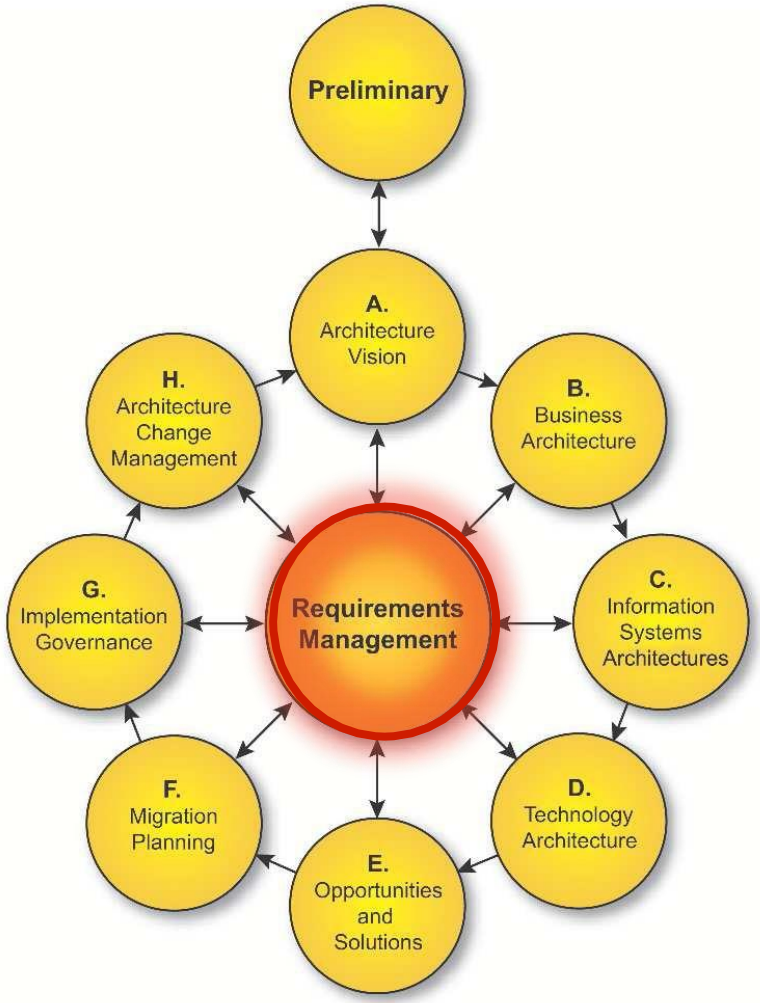
- Types of changes
  - Simplification change: via change management
  - Incremental change: via change management /partial re-architecting
  - Re-architecting change: via AMD



Requirements Management

Requirements Management examines the process of managing architecture requirements throughout the ADM

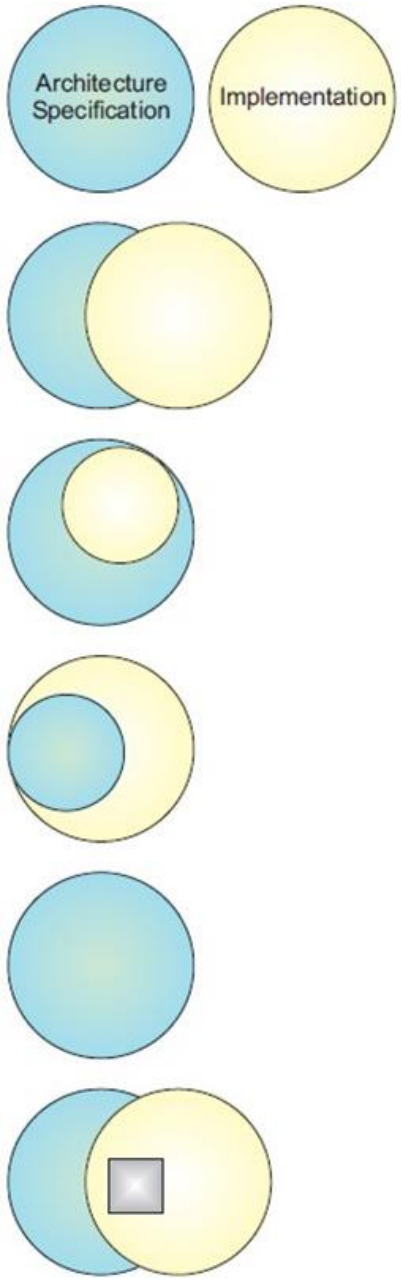
- Prioritization of Requirements
- Changes requirement/priority → Request Impact Statement/Assessment



Overview of the ADM

- **The ADM is a generic method**: It can be used by enterprises in a wide variety of different geographies, applied in different vertical sectors/ industry types and can be tailored to specific needs
- **The ADM is iterative**: Over the whole process, between phases, and within phases
- For each iteration of the ADM, a fresh decision must be taken as to:
  - The level of detail to be defined
- The architectural assets to be used:
  - Assets created in previous iterations of the ADM cycle within the enterprise
  - Assets available elsewhere in the industry (other frameworks, systems models, vertical industry models, etc.)
- To integrate the ADM with another enterprise framework
  - PRINCE2, ITIL, COBIT, CMMI, Zachman Framework

Compliance levels



Irrelevant:
The implementation has no features in common with the architecture specification (so the question of conformance does not arise).

Consistent:
The implementation has some features in common with the architecture specification, and those common features are implemented in accordance with the specification. However, some features in the architecture specification are not implemented, and the implementation has other features that are not covered by the specification.

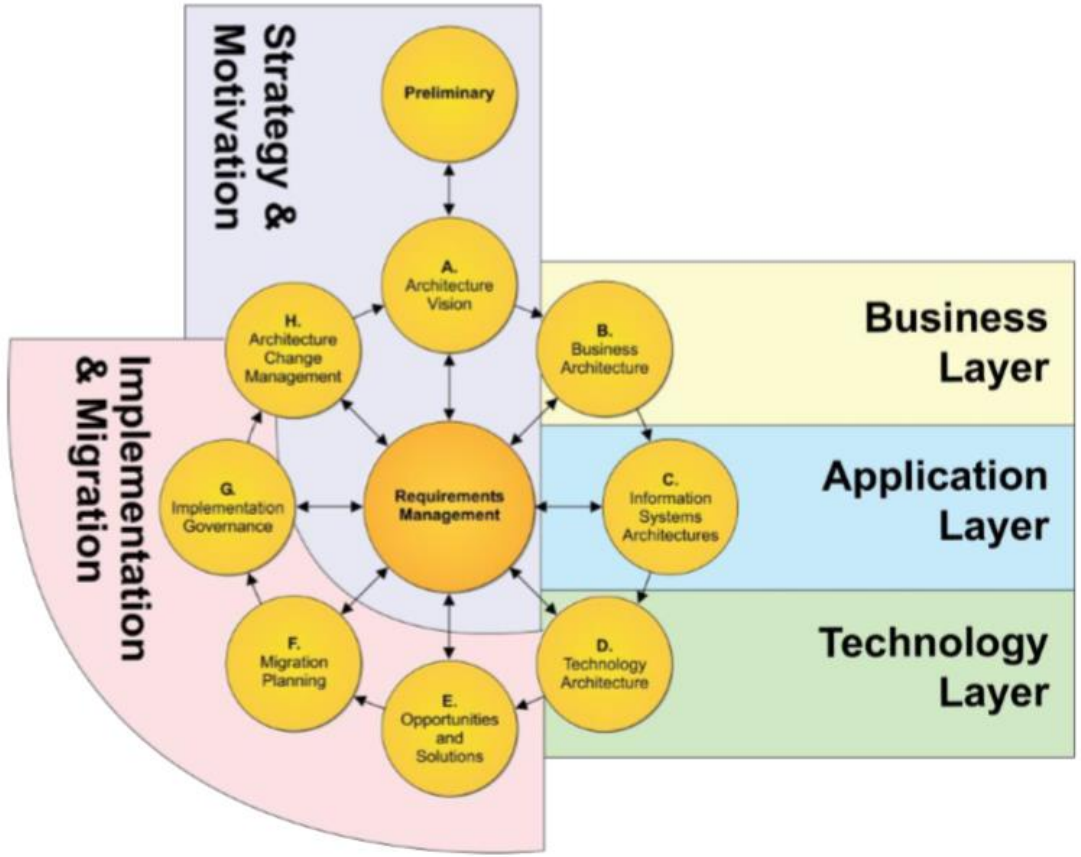
Compliant:
Some features in the architecture specification are not implemented, but all features implemented are covered by the specification, and in accordance with it.

Conformant:
All the features in the architecture specification are implemented in accordance with the specification, but some more features are implemented that are not in accordance with it.

Fully Conformant:
There is full correspondence between architecture specification and implementation. All specified features are implemented in accordance with the specification, and there are no features implemented that are not covered by the specification.

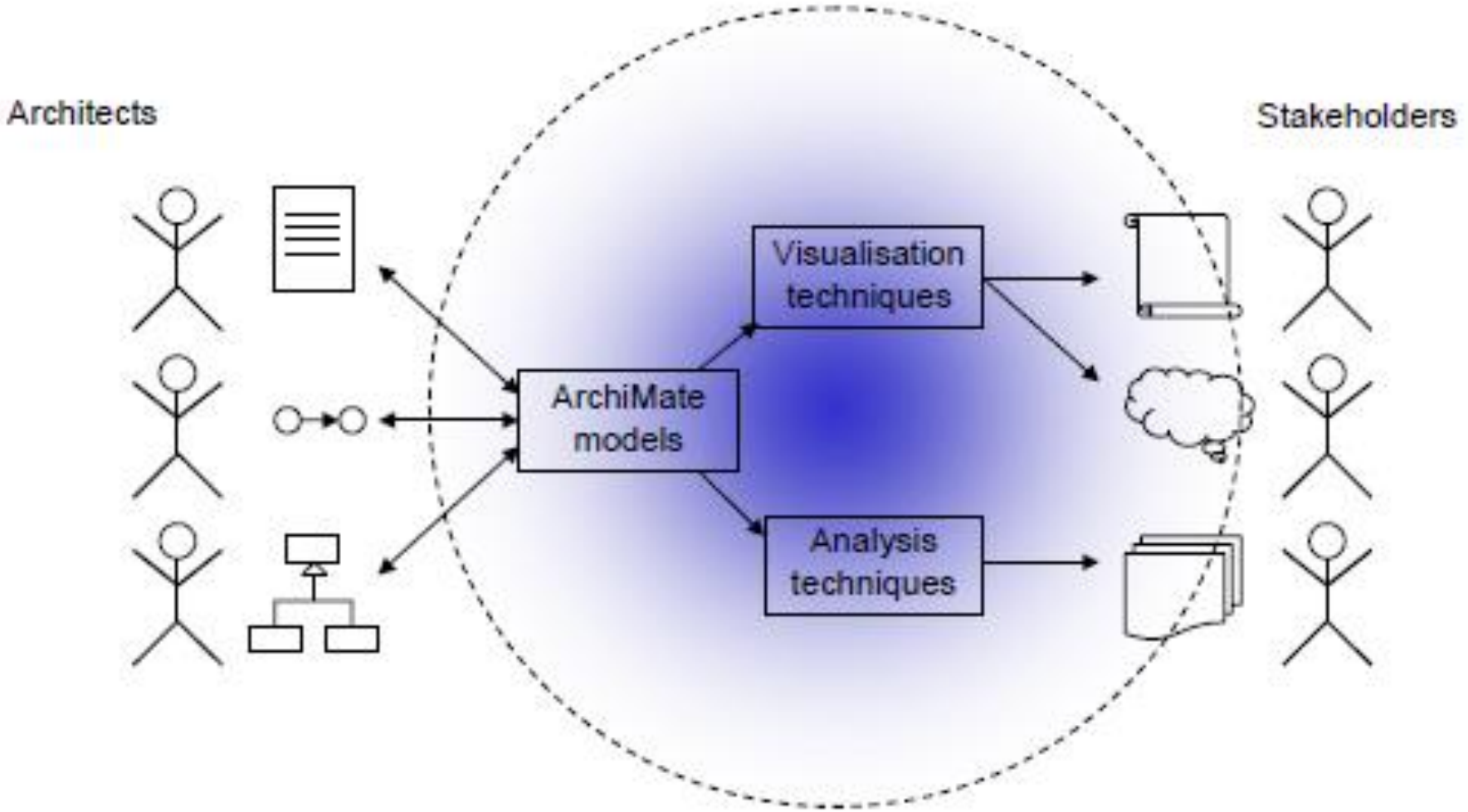
Non-conformant:
Any of the above in which some features in the architecture specification are implemented not in accordance with the specification.

Correspondence between ArchiMate and TOGAF



The Relationship between the ArchiMate Language and the TOGAF ADM

ArchiMate – why to use

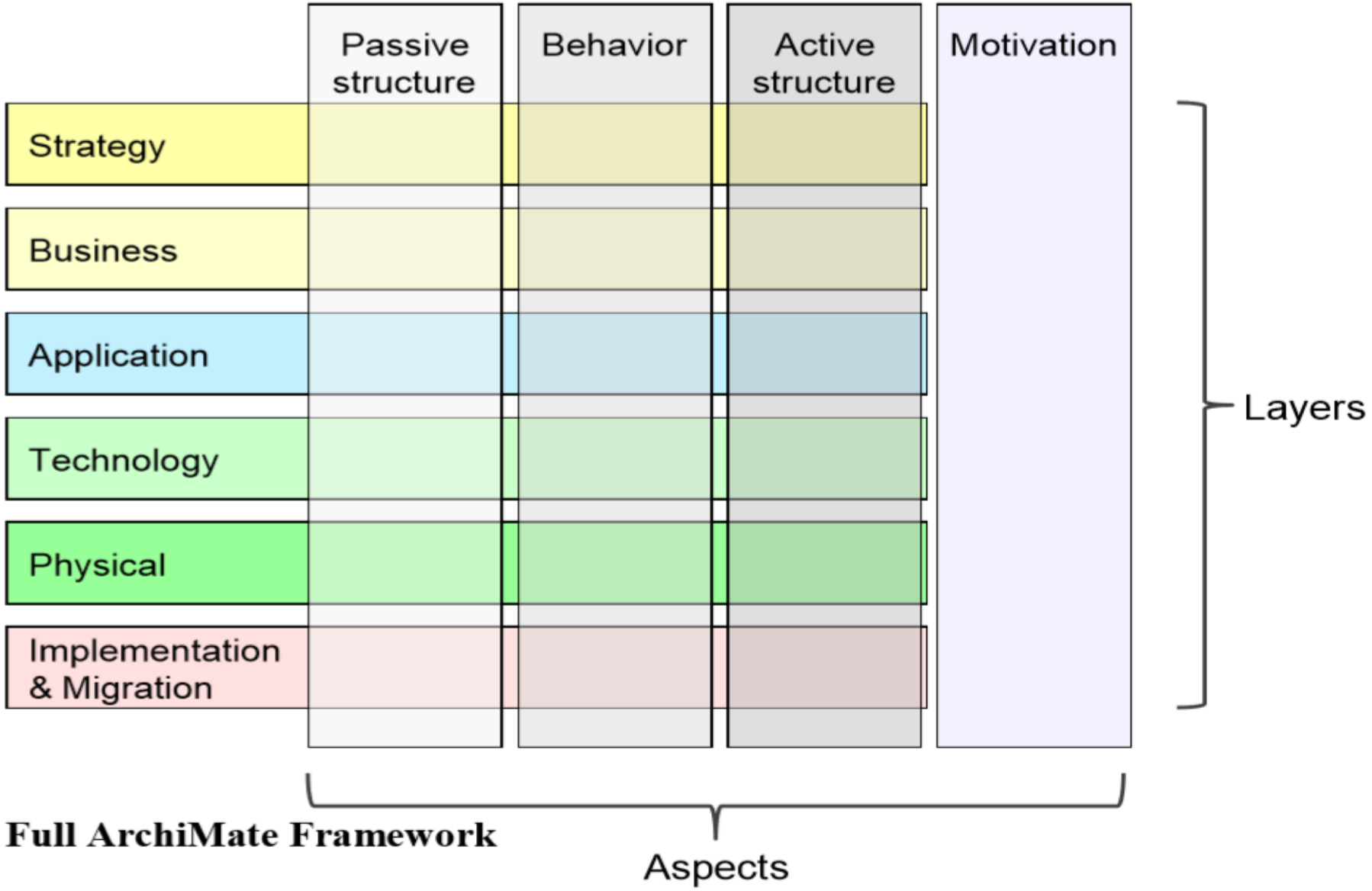


ArchiMate framework

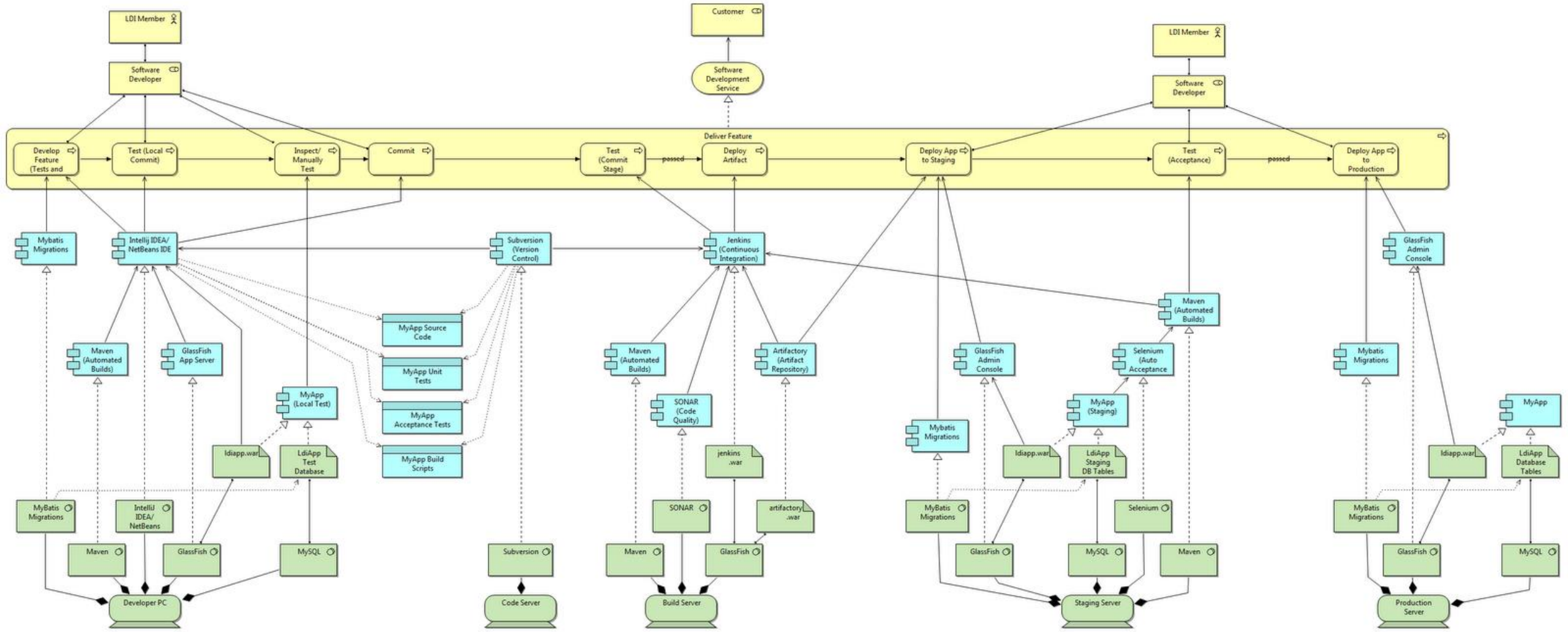
The dimensions of the framework are as follows:

- Layers: The three levels at which an enterprise can be modeled in the ArchiMate language Business, Application, and Technology (as described in Section 2.2).
- Aspects:
  - The *Active Structure aspect*, which represents the structural elements (the business actors, application components, and devices that display actual behavior; i.e., the "subjects" of activity).
  - The *Behavior aspect*, which represents the behavior (processes, functions, events, and services) performed by the actors. Structural elements are assigned to behavioral elements, to show who or what displays the behavior.
  - The *Passive Structure aspect*, which represents the objects on which behavior is performed. These are usually information objects in the Business Layer and data objects in the Application Layer, but they may also be used to represent physical objects.

ArchiMate framework



Full ArchiMate Framework


LDI Member
Software Developer
Customer
Software Development Service
LDI Member
Software Developer
Deliver Feature
Develop Feature (Tests and
Test (Local Commit)
Inspect/ Manually Test
Commit
Test (Commit Stage)
passed
Deploy Artifact
Deploy App to Staging
Test (Acceptance)
passed
Deploy App to Production
Mybatis Migrations
Intellij IDEA/ NetBeans IDE
Subversion (Version Control)
Jenkins (Continuous Integration)
Maven (Automated Builds)
GlassFish Admin Console
Maven (Automated Builds)
GlassFish App Server
MyApp Source Code
MyApp Unit Tests
MyApp Acceptance Tests
MyApp Build Scripts
Maven (Automated Builds)
SONAR (Code Quality)
Artifactory (Artifact Repository)
GlassFish Admin Console
Selenium (Auto Acceptance
MyApp (Staging)
Mybatis Migrations
Mybatis Migrations
MyApp
MyApp (Local Test)
Idiapp.war
LdiApp Test Database
jenkins .war
Idiapp.war
LdiApp Staging DB Tables
Idiapp.war
LdiApp Database Tables
MyBatis Migrations
IntelliJ IDEA/ NetBeans
SONAR
artifactory .war
MyBatis Migrations
Selenium
MyBatis Migrations
Maven
GlassFish
MySQL
Subversion
Maven
GlassFish
GlassFish
MySQL
Maven
GlassFish
MySQL
Developer PC
Code Server
Build Server
Staging Server
Production Server

Archi – Open Source ArchiMate Modelling

- An open source modelling toolkit to create ArchiMate models and sketches. Used by thousands of Enterprise Architects throughout the world.
- The Archi® modelling toolkit is targeted toward all levels of Enterprise Architects and Modellers.
- It provides a low cost to entry solution to users who may be making their first steps in the ArchiMate modelling language, or who are looking for an open source, cross-platform ArchiMate modelling tool for their company or institution and wish to engage with the language within a TOGAF® or other Enterprise Architecture framework.

Archi 4.3

- Multiplatform software for Enterprise Architechture modelling
- Download page: https://www.archimatetool.com/download/