Module 3 : Requirement Management : Central Phase of ADM

How to Proceed :

A. Part 1 : First read the immediate portion, which is a Summary portion : Part 1 is for reading right now

   This portion is important for

   1. Understanding TOGAF for practical purposes – Supplement class session understanding with this
   2. For Certification purposes, Level 1 and Level 2

B. Part 2 : Go through and workout the exercises in the Part 2 : Module4Questions&Answers. Very helpful for Certification preparation

C. Part 3 : Later when you find more time, do go through portion which says Part 3 : Detailed Courseware. That portion is useful for getting extra grades in Certification and for more proper understanding of TOGAF. Some sections of it are quoted from internet sources and from good authors as discovered by our Participants in earlier courses.

   In this Part 3, Case Study and its boxes with samples are for understanding purpose only.

Part 1 : Summary portion

You may like to first read this Quick Look : Glossary and Acronym

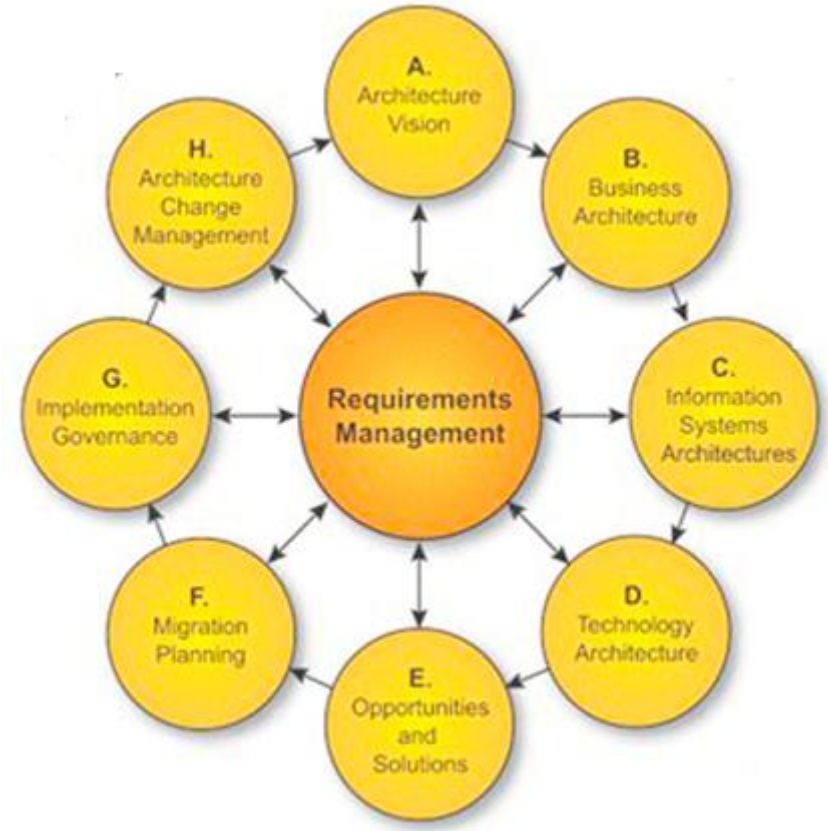
Requirements Repository : A Store managed by the Requirement Management Executive. Architectural Building Blocks made up of Artifacts are stored here, till the EA decides to move the matured ones to the Architecture Repository. Anything in the Requirement Repository can at best be re-used by the same Project Portfolio only, since it may undergo changes till it reaches a matured and stable sate.

Routing the Requirement : the necessary documents or artifacts related to specific requirements are moved from the Requirement Repository to an Architect who is working on it, and then moved back to the Repository when EA accepts that portion (version) of work. It is also possible that some requirements could be dropped in the process. The job of the Requirement Management Executive to keep track of these happenings and route it – send it to the right person in the EA department at the right time as scheduled.

Dynamic Forever Process : This Architectural process never stops. When a Project has reached completion towards end of Phase G, changes therein may come up to trigger the next major release to be taken up again in the ADM

Priorities : Relative priority of which portion of the work is to be taken in preference to another one. The actual timeline of work is based on priority and the Roadmap for the Portfolio.

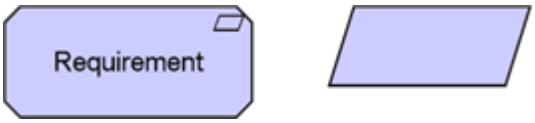
Impact : Even within an ADM cycle, the architectural; work done at every stage is to be checked for possible impact from other areas of B, D, A, and T - with the work done in previous phases. The document in this connection is known as Requirements Impact Assessment



Requirement

A Requirement represents a statement of need that must be met by the Architecture

In the end, a business goal must be realized by a plan or concrete change goal, which may or may not require a new system or changes to an existing system



In TOGAF ADM,

Requirement is captured in Phase A

Then consolidated as ABB – Architectural Building Blocks, in Phases B, C and D

Thereafter made into SBB – (Service Centric) Solution Building Blocks in other Phases and gets implemented till Phase G

Requirement : Statement of Architecture need

Constraint

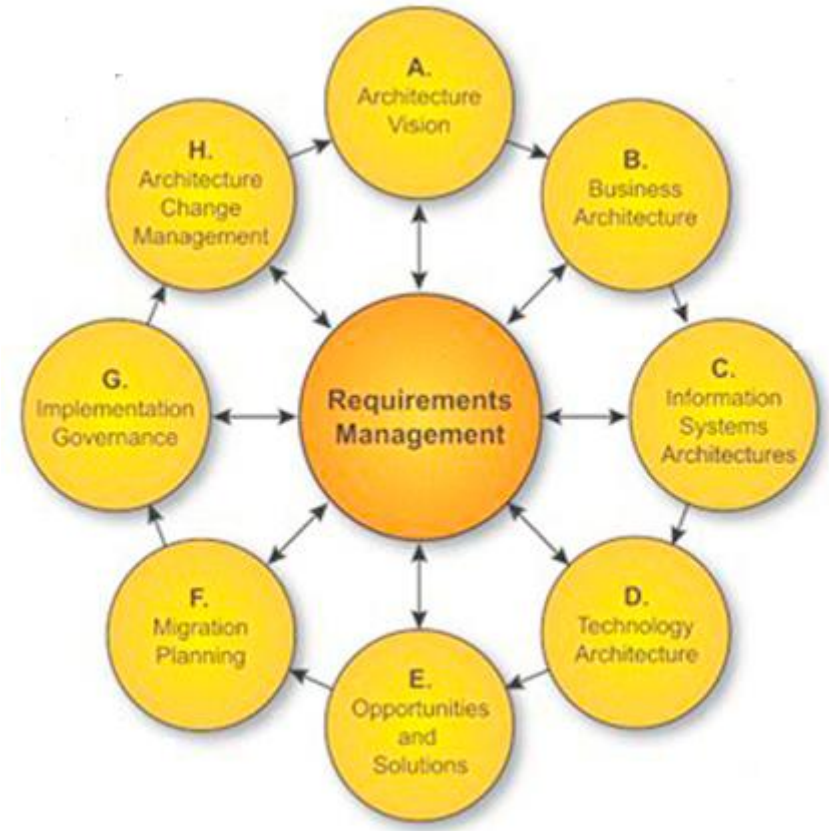
A Constraint represents a factor that prevents or obstructs the realization of goals

Constraints show the limits within which the system should be realized.

Constraint : Factor that prevents goal realization

The Requirements Management activity of TOGAF ONLY **defines a process of handling the requirements**

during Architecture development through ADM



It **does not generate any Requirement**.

Just handles the Requirements which are generated, mainly in Phases A to D or in later Phases

ADM Requirements Management

- The process of **managing architecture requirements**
  - applies to all phases of the **ADM cycle**
  - is **central** to the ADM process
- is a dynamic process addressing the **identification of requirements**, their storage and delivery to the phases

Req Mgt never stops
Why ?

Requirements Development

- **First high level requirements** are developed in the Architecture Vision
- For each ADM phase, from **Preliminary to Phase H**

Start

  - **Select the approved requirements** for that phase as held in the Requirements Repository and Architecture Requirements Specification

Continues

  - At the completion of a phase the status of all such requirements needs to be updated
- During phase execution
  - **New requirements** generated for future architecture work within the scope of the current Statement of Architecture Work need to be documented within the **Architecture Requirements Specification**
  - New requirements which outside of the scope of the current Statement of Architecture Work must be input to the Requirements Repository for management through the Requirements Management process

Newer additions also

Requirement is **never generated** in this Phase

Requirement Management Phase "manages" the journey in and out of ADM Phases

ADM Phases work on Requirements, Architecture creation and so on

Requirement is captured in Phase A

Then consolidated as ABB – Architectural Building Blocks, in Phases B, C and D

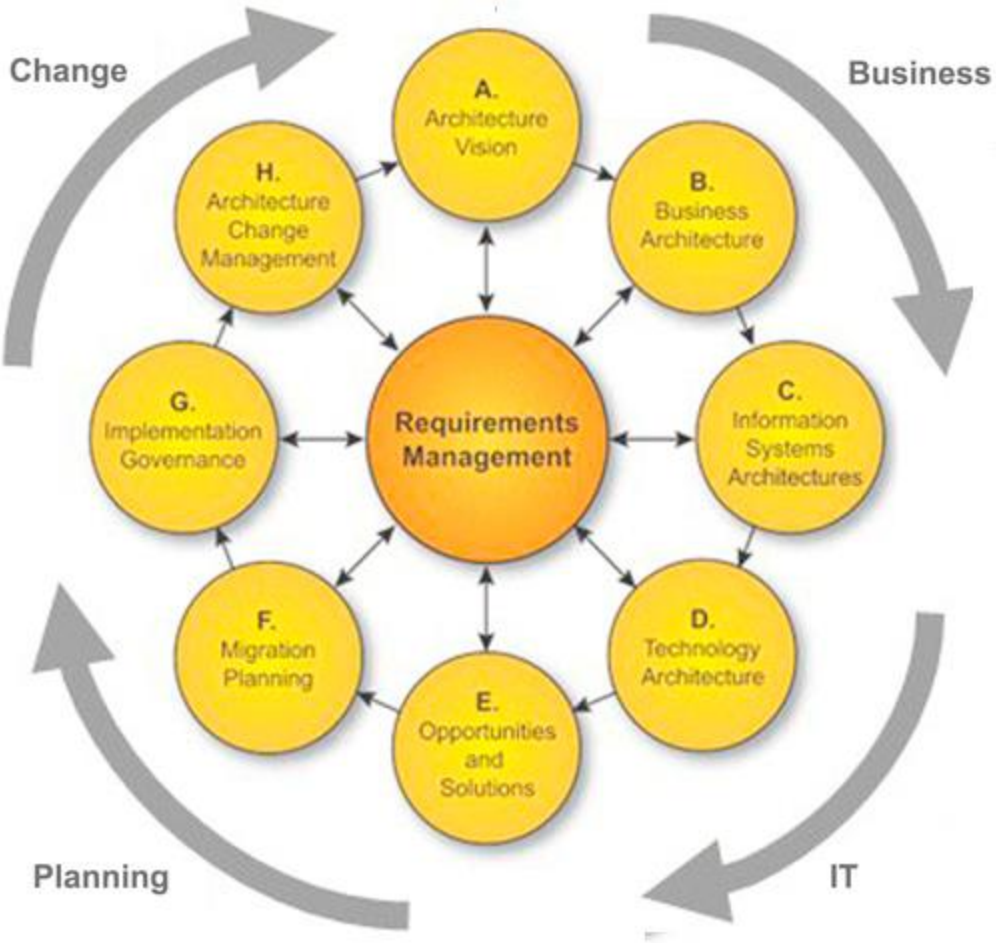
Thereafter made into SBB – (Service Centric) Solution Building Blocks in other Phases and gets implemented till Phase G

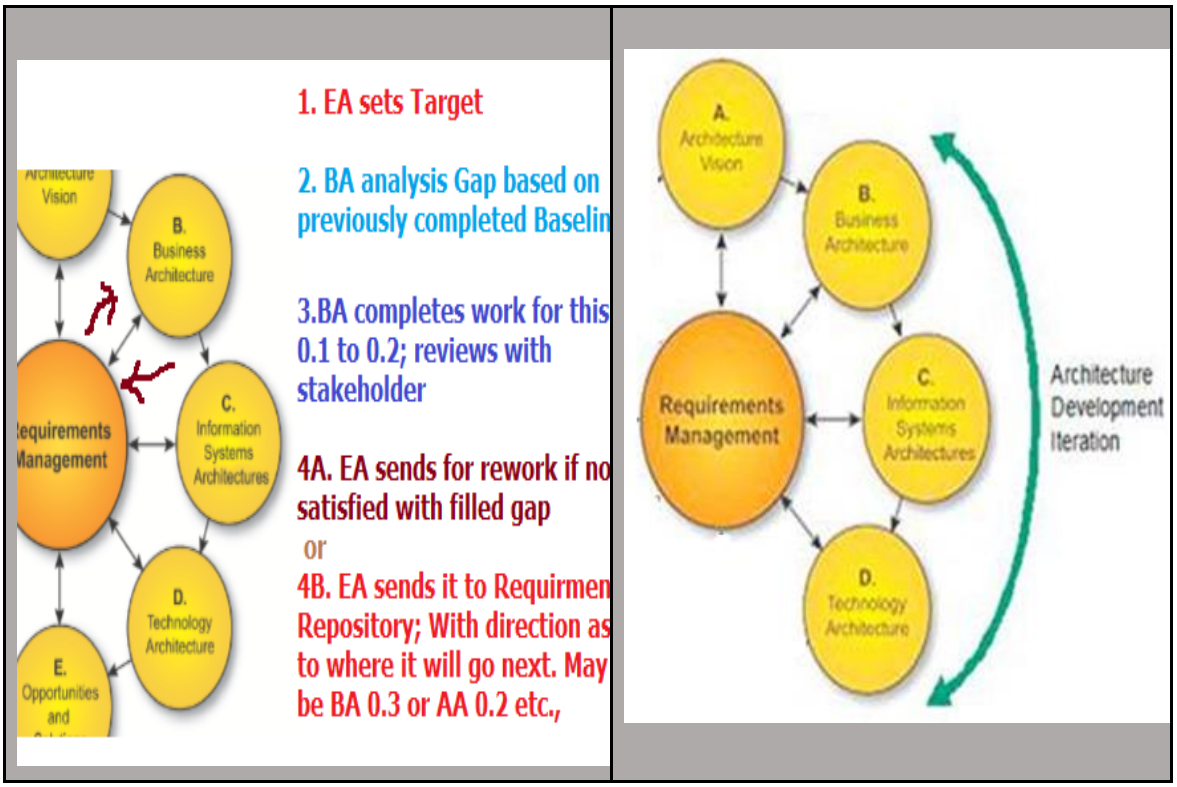
Usually, Phase B firms up the Requirement based on business need (as per Vision Phase) and this is done after intensive interaction with the stakeholders. It is generated in line with the aspirational Vision of Phase A

Hereafter the Requirements moves into various Phases, as directed by the EA. Architectures are generated and Requirements are firmed up

This is the **ONLY Phase in ADM where no Building Blocks are generated**. But this Phase still **'manages'** – Stores, sends it at right time for next action and carries out all such 'management'

Manages movement of Requirement document and the Architectural BB work done on it





The objectives of the Requirements Management Phase are to :

- Ensure that the Requirements Management process is sustained and operates for all relevant ADM Phases

  Requirement generation and addressal is never stopped

Requirement Management : Sustain

- Manage Architecture Requirements identified during any execution of the ADM cycle or a Phase

  Requirements accepted are taken through ADM so that they become "Realization" at a later point of time

ADM Inputs and Outputs

- TOGAF defines a number of inputs and outputs for each phase **It is in the Content Frame**
  - These are suggestions, therefore **not mandatory**
  - Output of an early phase may be modified in a later phase **See arrows flowing between A to H Phases, but always ROUTED through the Requirment Management Phase**
  - Version numbers are used to manage the output
  - A convention is used to illustrate the evolution of deliverables **Ideally at ABB Segment Levels**
    - 0.1 – a high level outline deliverable **Vision Phase**
    - 1.0 – a formally reviewed detailed deliverable **End of Phase D**

"Manage" the Requirement

Not to create it or approve it

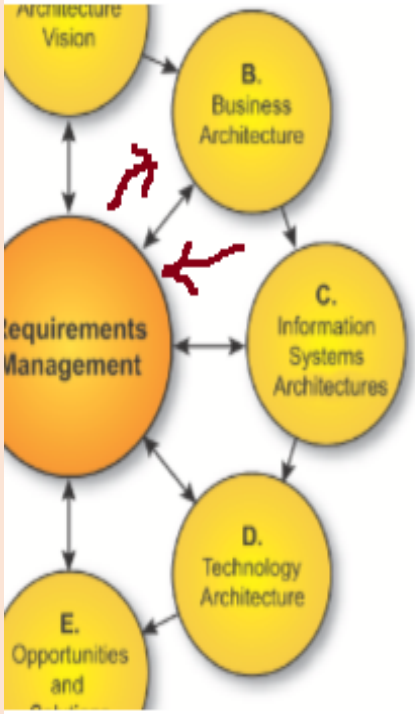
Just a Routing mechanism

Route the Requirement

Send it to the next Phase of work

Just a Routing mechanism

Routing is :



1. EA sets Target

2. BA analysis Gap based on previously completed Baseline

3.BA completes work for this 0.1 to 0.2; reviews with stakeholder

4A. EA sends for rework if not satisfied with filled gap

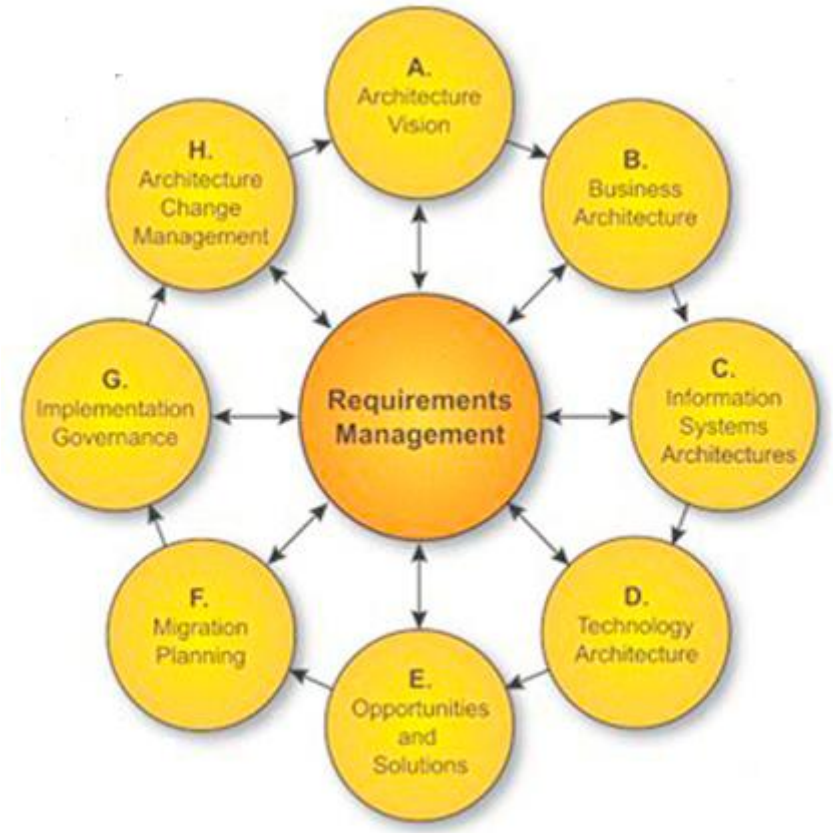
or

4B. EA sends it to Requirment Repository; With direction as to where it will go next. May be BA 0.3 or AA 0.2 etc.,

Occupies Central position in the ADM cycle.

Actually, Requirements Management is

more a permanent activity than a Phase



Permanent Activity

Connects to almost all Phases

Even when Project reaches completion

Permanent Activity

Dynamic Forever Process

The versioning of output in Phases : managed through version numbers.

A version numbering convention 

is used within the ADM to illustrate the evolution of Baseline and Target Architecture Definitions.

Examples : Business Architecture version 0.1; Business Architecture version 1.0

Version **0.1** indicates that a high-level outline of the Architecture is in place.

Version **1.0** indicates a formally reviewed, detailed Architecture.

Can use numbers in between to indicate architectured progress

Why version numbers ? A Baseline is a stable, change-controlled configuration of work, used for release planning or other delivery milestones in a project. The discipline should start early, with Business Architecture downwards.

Enterprise Architects also decide priority of work, from time to time
The Roadmap and priority therein is recorded in Requirements Repository

WIP : Work In Progress

Functional Requirement : From Business Scenario

NFR – Non Functional Requirement

– Arises out of Architectural outlook

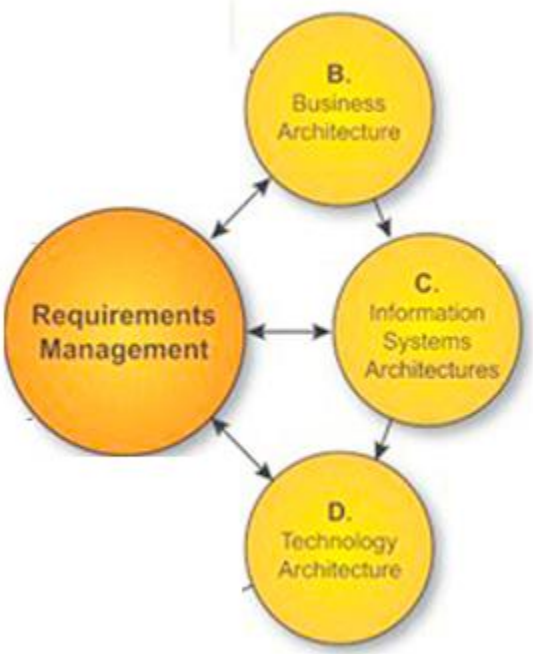
The move towards Requirement changes :

Are recognized at highest level in Phase A : Architecture Vision with initial 0.1 slice of Architectural work

Are prepared as 'understandable' ABBs – Architecture more than solution stage in Phases B to D - As version slices - 0.2, 0.3 and so on till 1.0

Every such version slice and the Architecture work carried out in Phases B to D can influence each segment of B D A T, after its due temporary storage in Requirement Repository mainlined by the Requirement Management Phase.

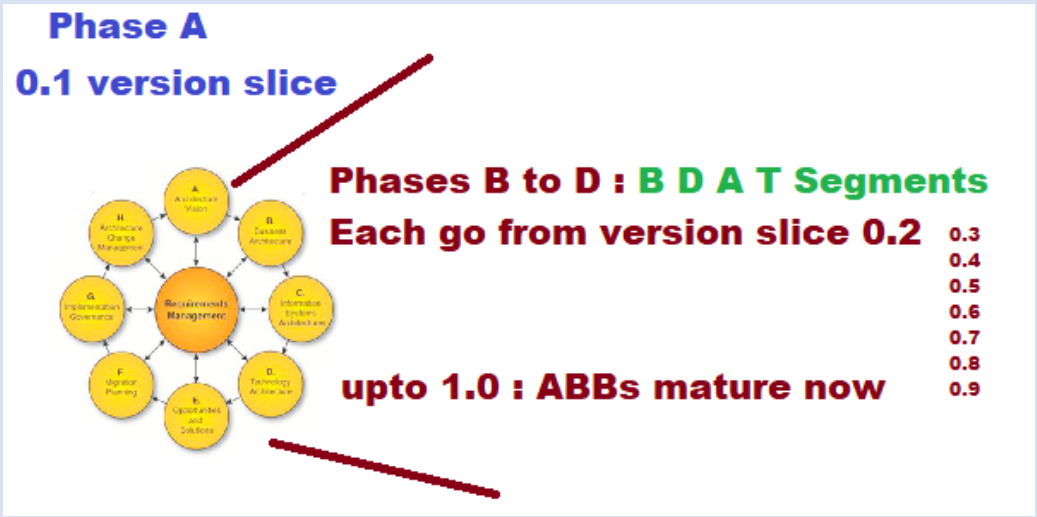
That explains why there are lines connecting Phase B to D to the central Phase :



WIP : Work In Progress

The Architecture Requirements Specification provides a set of quantitative statements that outline what an implementation project must do in order to comply with the Architecture.

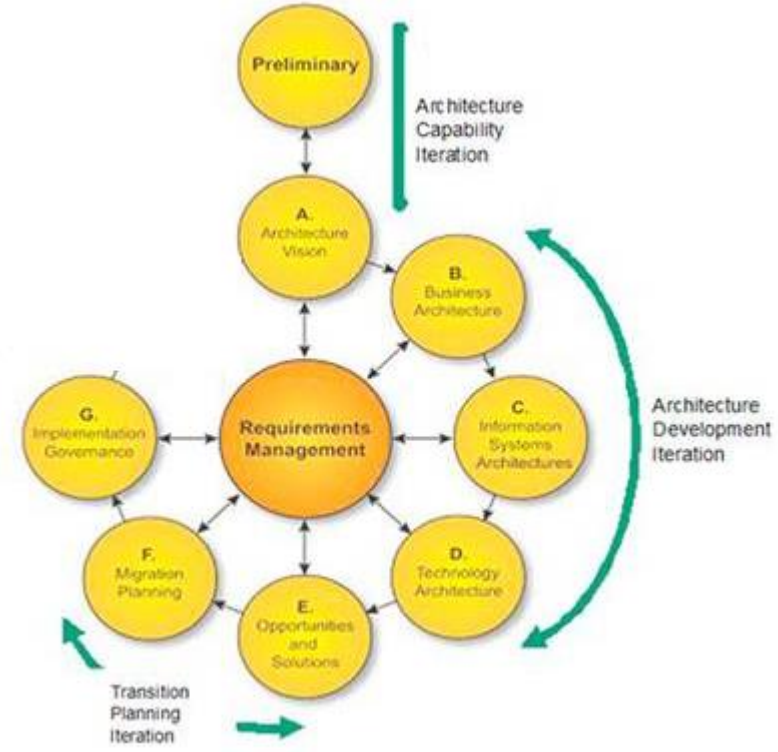
An Architecture Requirements Specification will typically form a major component of an implementation contract or contract for more detailed Architecture Definition.



They become 'implementable' SBBs – Solution Building Blocks in Phase E

Final decision based on roadmap and other criterion is made in Phase F

Actual implementation takes place parallel to Phase G



Steps

Requirements Management Steps	ADM Phase Steps
	1. Identify/document requirements
2. Baseline requirements	
3. Monitor baseline requirements	
Not changes in what is coded	
	4. Identify changed requirement
They mean a fresh requirment	over earlier one already accepted
5. Identify changed requirement and record priorities	
	6. Assess impact of change
Requirment Impact Assessment	
	7. Implement changes arising from Phase H
Keep change list eady for next ADM	
8. Update the Requirements Repository with the changes	
9. Implement change in the current phase	
Urgent changes happen in current ADM	- Radical, Sudden : Cannot wait
10. Assess and revise gap analysis for past phases	

In actual fact, this requirement contains two needs of different types :

- A **functional requirement** : "The client must be able to order a product online."
- A **nonfunctional requirement** (**NFR**) : "It must be possible to place an order at all times, 24 X 7 X 365"

The functional handles the "what," while the nonfunctional deals with the conditions under which the service is provided.

These conditions concern **performance, security, availability, reliability**, and so on and are the object of detailed listings.

The Requirement Repository is kept up-to-date during Phases B to D and thereafter too

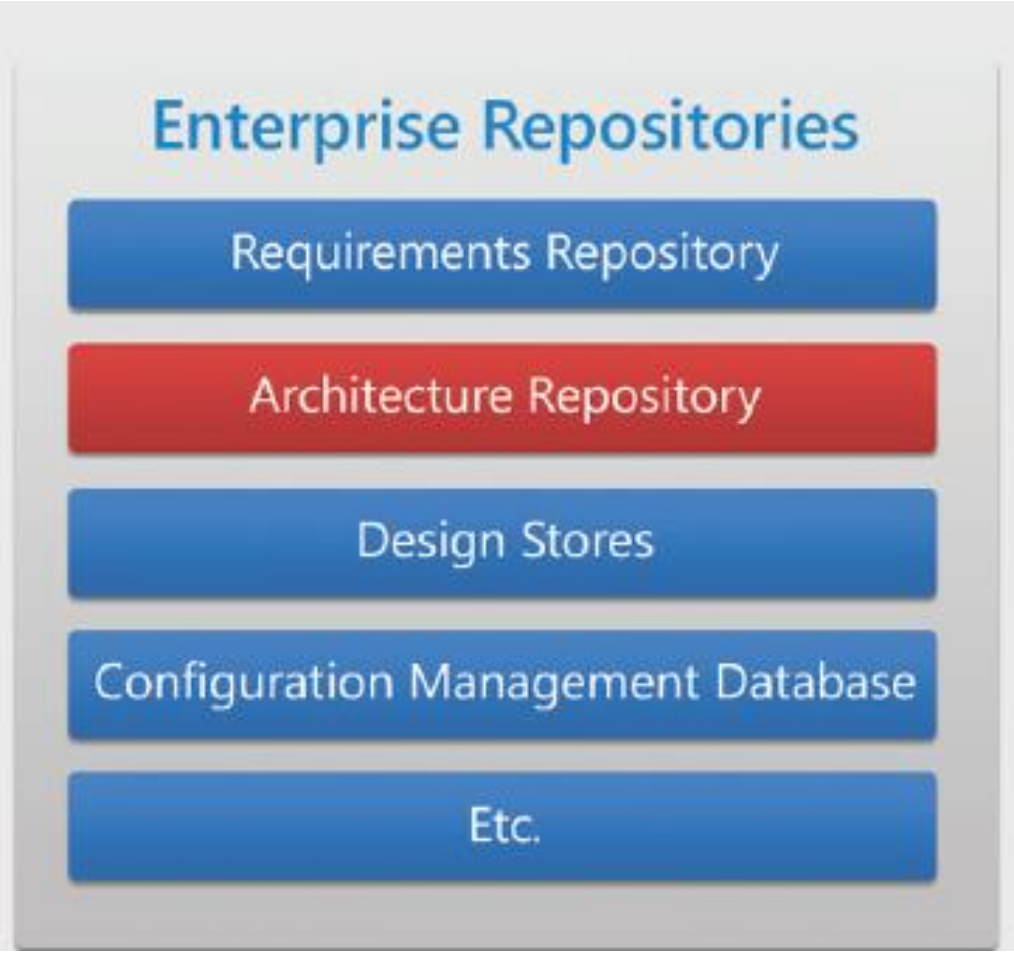
Update the Requirements Repository

with information relating to the changes requested, including stakeholder views affected.

Update the Architecture Requirements Repository with information relating to the Requirements due to Impact and Stakeholder needs as requested, including stakeholder views affected

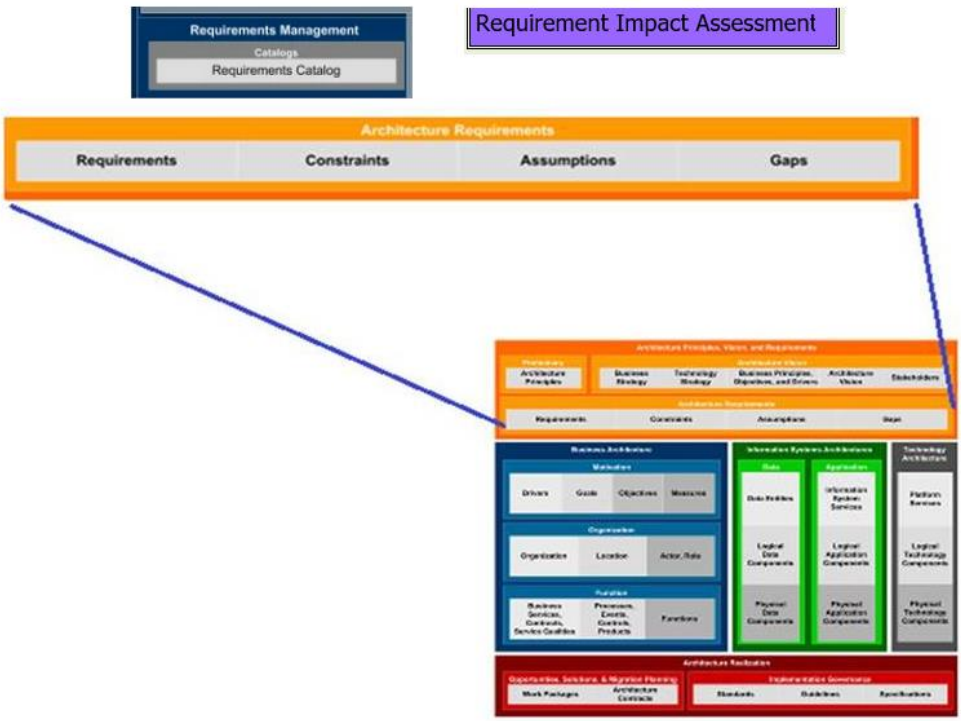
Physically managed by Document Executive

Artifacts and supporting documents relating to every Project is kept in Requirements Repository during the Work-in-Progress stages. They move to Architecture Repository only on completion of all Architectural work relating to its overall Target.



Document related to this Phase but more used in Phase H :
Requirement Impact Assessment

Each Phase of ADM produces Artifacts that enriches the Architecture Content Framework. However, this Phase never produces any, but only captures and stores them during WIP – Work-In-Progress stages. For example :



Artifacts

THE Open GROUP

Refer to : 32.2.18 Requirements Impact Assessment in TOGAF 9.2 Documentation

Artifacts '**managed**' in Requirements Mgt Phase …

Requirements Constraints Assumptions Gaps

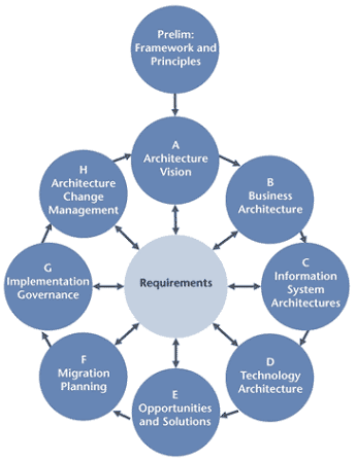
As a Requirements Catalog

Document produced in Requirements Mgt Phase

Requirement Impact Assessment

Think : Every other Phase in ADM has beginning and end point. What about the Requirement Management Phase ?

Answer : This is dynamic, forever Central Phase. As long as Enterprise Architecture work is going on (which is forever), this Phase will linger on



A RELATED QUESTION, IN LEVEL 1

:  301

Q : The Requirements Management Phase is responsible for **which one** of the following activities ?

A. Addressing requirements
B. Disposal of resolved requirements
C. Generating requirements
D. Managing the flow of requirements
E. Prioritizing requirements

Answer : D Name itself is Requirement Management Phase. Not Requirement Solution Phase

Summary of the steps of Central Phase : Requirement Management

(More clearly, many are related to steps of other Phases and with the role of the Architects, but a progress control is recorded for proper routing of them by the Requirement Management Executive in this Phase)

Document the Requirements - use business scenarios, or an analogous technique : Note that capturing the Requirement by engaging with the Stakeholders is not done by this Phase. It is part of steps in other Phases

Look into the Baseline to Determine and Record the Priorities : Storing them in Requirement Repository

Continue the process of monitoring the Baseline Requirements so that fresh Targets are achieved and they become Baseline for next version slice

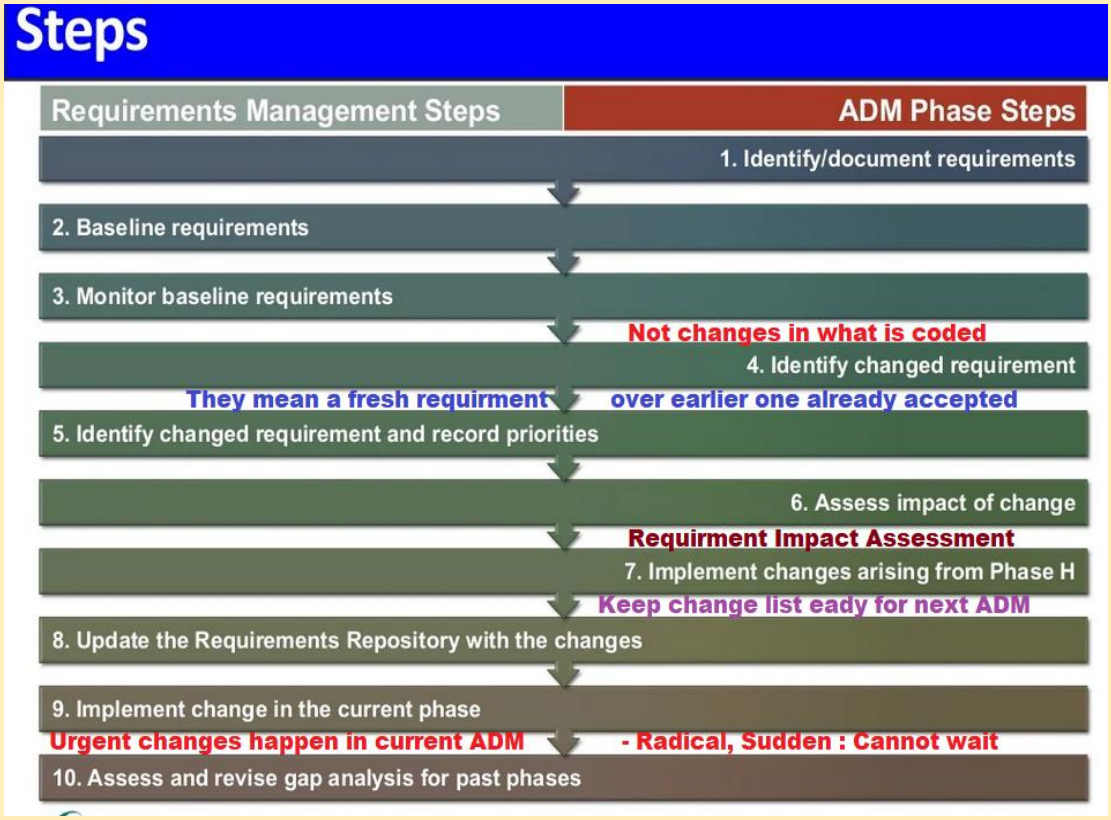
Be alive to change in Requirements. Would need change in priorities as well as change in documentation

Do record the decisions related to the requirements as and when the Gap Analysis and Building Block preparation work is performed in respective Phases

Record the result of assessment relating to impact of every Building Block in Requirements Impact Statement

Update the Architecture Requirements Repository with information relating to the Requirements due to Impact and Stakeholder needs as requested, including stakeholder views affected

Implement change in the current Phase : Means, role of Requirement Management is to keep track of such a progress



Part 2 : Module 3 Questions and Answers

(Also Explanations)

Please answer questions appearing below on a piece of paper and then check the answer and explanation appearing immediately below the questions. Some Questions may be on earlier modules too.

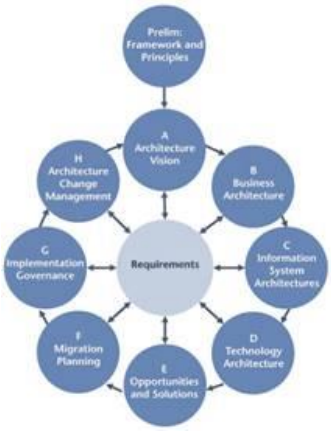
You can choose the ones you want to answer now and keep the rest for a workout on your own later on.

The pictures that appear next to the question is only to break the monotony and has no special meaning.

The star rating gives you a clue of the relative importance of questions, from Certification viewpoint. Three-star questions may appear more often than two star and so on

Think : Every other Phase in ADM has beginning and end point. What about the Requirement Management Phase ?

Answer : This is dynamic, forever Central Phase. As long as Enterprise Architecture work is going on (which is forever), this Phase will linger on



301 The Requirements Management Phase is responsible for which one of the following activities ?

A. Addressing requirements
B. Disposal of resolved requirements
C. Generating requirements
D. Managing the flow of requirements
E. Prioritizing requirements

Answer : D

Explanation :

"Managing" in this context is not to mean

a) Generation of Requirement which is mostly in phase B, based on high level Vision of Phase A
b) Addressing and Disposing which are done in Phases such as B, C, D
c) Setting Priority which is done by EA

See : **16.5 : Approach (to Requirement Management Phase)**

Note also that the Requirements Management process itself does not dispose of, address, or prioritize any requirements; this is done within the relevant phase of the ADM. It is merely the process for managing requirements throughout the overall ADM.


302

Which phase of the ADM is an on-going activity that is visited throughout a TOGAF architecture project ?
A. Architecture Change Management
B. Implementation governance
C. Migration planning
D. Preliminary Phase
E. Requirements Management

Answer : E

Explanation :

Requirements Management is shown as a central Phase and is an on-going activity that is visited throughout a TOGAF architecture project.

See : **16.1 Objectives (of Requirement Management Phase)**

The objectives of the Requirements Management phase are to :

■ Ensure that the Requirements Management process is sustained and operates for all relevant ADM phases

Also See : **16.5.1 General**

As indicated by the "Requirements Management" circle at the centre of the ADM graphic, the ADM is continuously driven by the Requirements Management process.

303 Which of the  following statements about Requirements Management Phase is correct ?

A. This phase is executed at the beginning
B. It focuses on minimizing changes to requirement
C. It does not dispose of, address, or prioritize any requirements; this is done within relevant phase of the ADM
D. TOGAF does not have a recommend ADM process for requirements management
E. Prioritizing requirement is an activity of this phase

Answer : C

Explanation :

Note from B, C, D Phases, the step : **Define Candidate Roadmap Components**

Following the creation of a Baseline Architecture, Target Architecture, and gap analysis, a …. roadmap is required to prioritize activities over the coming phases.

Scenario Based Question : SBR – 3001

Scenario is about : Automotive Company : Stakeholder Analysis and Requirement Management

You are serving as the Lead Enterprise Architect at a major supplier in the automotive industry. The company is headquartered in Cleveland, Ohio with manufacturing plants across the United States, Brazil, Germany, Japan and South Korea. Each of these plants has been operating its own planning and production scheduling systems, as well as custom developed applications that drive the automated production equipment at each plant.

The company is implementing lean manufacturing principles to minimize waste and improve the efficiency of all of its production operations. During a recent exercise held for internal quality improvement, it was determined that a significant reduction in process waste could be achieved by replacing the current planning and scheduling systems with a common Enterprise Resource Planning (ERP) system located in the Cleveland data centre.

This central system would provide support to each of the plants replacing the functionality in the existing systems. It would also eliminate the need for full data centers at each of the plant facilities. A reduced number of IT staff could support the remaining applications. In some cases, a third-party contractor could provide those staff.

The Enterprise Architecture department has been operating for several years and has mature, well developed architecture governance and development processes that are strongly based on TOGAF 9.

At a recent meeting, the Architecture Board approved a Request for Architecture Work sponsored by the Chief Engineer of Global Manufacturing Operations. The request covered the initial

architectural investigations and the development of a comprehensive architecture to plan the transformation.

The Common ERP Deployment architecture project team has now been formed, and the project team has been asked to develop an Architecture Vision that will achieve the desired outcomes and benefits. Some of the plant managers have expressed concern about the security and reliability of driving their planning and production scheduling from a remote centralized system. They expect many incremental changes would evolve as the project progresses. The Chief Engineer wants to know how these concerns can be addressed.

During the initial meeting of the Common ERP Deployment architecture project team, a number of alternative recommendations for how to proceed are put forward by members of the team.

You have been asked to select the most appropriate recommendation to ensure that the team evaluates different approaches to the problem and clarifies the initial requirements and the path for the architecture. Thereafter the Requirement Management Phase is to be followed strictly as per TOGAF.

Based on TOGAF 9, which of the following is the best answer ?

A. The team should develop Baseline and Target Architectures for each of the manufacturing plants, ensuring that the views corresponding to selected viewpoints address key concerns of the stakeholders.

The baseline requirements should be monitored at the end of the ADM cycle. The changed Requirements is to be identified and recorded. The impact thereon is to be studied at the end of the ADM cycle. The Requirement Repository is updated at this stage.

A consolidated gap analysis between the architectures will then be used to validate the approach, and determine the capability increments needed to achieve the target state. This is what is done at the end of the ADM cycle.

B. The team should exercise due diligence and carefully research vendor literature and conduct a series of briefings with vendors that are on the current approved supplier list. Based on the findings from the research, the team should define a preliminary Architecture Vision. The team should then use that model to build consensus among the key stakeholders.

The baseline requirements should be monitored throughout the ADM cycle. No changes in Requirement is to be allowed, as the project cycles through ADM. Change Request at best is to be identified and recorded after due priority for it is assigned. The impact of proposed change is to be studied at the last Phase of ADM.

C. The team should use stakeholder analysis to understand who has concerns about the initiative.

The team should then hold a series of interviews at each of the manufacturing plants using the business scenario technique. This will then enable them to identify and document the key high-level stakeholder requirements for the architecture.

The baseline requirements should be monitored throughout the ADM cycle. The changed Requirements, as the project cycles through ADM is to be identified and recorded after due priority for it is assigned. The impact thereon is to be studied at each Phase of ADM. The Requirement Repository is kept updated throughout these progressions.

D. The team should conduct a pilot project that will enable vendors on the short list to demonstrate potential solutions that will address the concerns of the stakeholders. Based on the findings of that pilot project, a complete set of retirements will be developed that will drive the evolution of the architecture.

Hand holding clues and approach tips from Faculty :

Issues in focus :

Need to minimize waste and improve the efficiency in production operations; significant reduction in process waste could be achieved by replacing the current planning and scheduling systems with a common ERP System

How the changes in Requirement is to managed in the ADM cycle

Aims :

Concern about the security and reliability; they expect many incremental changes would evolve as the project progresses. The Chief Engineer wants to know how these concerns can be addressed.

Requirement Management Phase is to be followed strictly as per TOGAF

To do : Select the approach which accommodates all the issues of the problem and proceeds with the initial requirements and the path for the architecture

Chapter 16 : ADM Architecture Requirements Management

DO NOT PROCEED TILL IT IS OPEN IN A WINDOW IN YOUR SYSTEM

16. ADM Architecture Requirements Management

Chapter Contents

16.1 Objectives | 16.2 Inputs | 16.3 Steps | 16.4 Outputs | 16.5 Approach

A. The team should develop Baseline and Target Architectures for each of the manufacturing plants, ensuring that the views corresponding to selected viewpoints address key concerns of the stakeholders.

The baseline requirements should be monitored at the end of the ADM cycle. The changed Requirements is to be identified and recorded. The impact thereon is to be studied at the end of the ADM cycle. The Requirement Repository is updated at this stage.

A consolidated gap analysis between the architectures will then be used to validate the approach, and determine the capability increments needed to achieve the target state. This is what is done at the end of the ADM cycle.

Baseline and Target Architectures for each of the manufacturing plants – Wrong wording. Architectures are developed for the IT (software, hardware) portions as per projects and portfolios and not for a plant as a whole

The baseline requirements should be monitored at the end of the ADM cycle. – Actually they are monitored throughout ADM and not at the end

The changed Requirements is to be identified and recorded -True, and comes under Requirement Management, since this phase records them and routes them

The impact thereon is to be studied at the end of the ADM cycle – Actually they are monitored throughout ADM and not at the end

The Requirement Repository is updated at this stage - True, though these are also updated in subsequent Phases as and when Gaps are addressed progressively

B. The team should exercise due diligence and carefully research vendor literature and conduct a series of briefings with vendors that are on the current approved supplier list. Based on the findings from the research, the team should define a preliminary Architecture Vision. The team should then use that model to build consensus among the key stakeholders.

The baseline requirements should be monitored throughout the ADM cycle. No changes in Requirement is to be allowed, as the project cycles through ADM. Change Request at best is to be identified and recorded after due priority for it is assigned. The impact of proposed change is to be studied at the last Phase of ADM.

Team should define a preliminary Architecture Vision – True. Vision Phase starts ADM Architecture cycle whenever major Change Request is accepted.

The team should then use that model to build consensus among the key stakeholders – True. Stakeholder Engagement is closely related to Requirement Management

The baseline requirements should be monitored throughout the ADM cycle – True

No changes in Requirement are to be allowed, as the project cycles through ADM – Not true. Minor change requests are accepted at intermediate stages.

The impact of proposed change is to be studied at the last Phase of ADM – Actually they are monitored throughout ADM and not at the end

C. The team should use stakeholder analysis to understand who has concerns about the initiative.

The team should then hold a series of interviews at each of the manufacturing plants using the business scenario technique. This will then enable them to identify and document the key high-level stakeholder requirements for the architecture.

The baseline requirements should be monitored throughout the ADM cycle. The changed Requirements, as the project cycles through ADM is to be identified and recorded after due priority for it is assigned. The impact thereon is to be studied at each Phase of ADM. The Requirement Repository is kept updated throughout these progressions.

Use stakeholder analysis to understand who has concerns about the initiative -Important point in this situation and in all projects.

Hold a series of interviews at each of the manufacturing plants using the business scenario technique – Mention of Business Scenario, the most appropriate technique is to be noted

Identify and document the key high-level stakeholder requirements for the architecture – Important

Baseline requirements should be monitored throughout the ADM cycle -True. Baseline could change somewhat as we enter each Phase, especially in Phase E

The changed Requirements, as the project cycles through ADM is to be identified and recorded after due priority for it is assigned – Note that priority assignment is related to Requirement Management Phase

The impact thereon is to be studied at each Phase of ADM – True
Requirement Repository is kept updated throughout these progressions – Very important point for Requirement Management Phase

D. The team should conduct a pilot project that will enable vendors on the short list to demonstrate potential solutions that will address the concerns of the stakeholders. Based on the findings of that pilot project, a complete set of retirements will be developed that will drive the evolution of the architecture.

Should conduct a pilot project – TOGAF never suggests a pilot project. **Every point in this answer choice thus is seen to be irrelevant**

Answer :

Best answer : C : The only answer that fits in the steps of Requirement management Phase, apart from suggesting the stakeholder analysis, business scenario and high level stakeholder requirement document.

Second best Answer : B : More green points and less of red points

Third best Answer : A

D is the worst answer for obvious reasons

304 What TOGAF  deliverable identifies changes that are needed to the current architecture requirements and specification, and also documents the implications of change ?

A. Architecture Vision
B. Requirements Impact Assessment
C. Gap Analysis Results
D. Architecture Landscape
E. Architecture Roadmap

Answer : B

Explanation :
Throughout the ADM, new information is collected relating to an architecture. As this information is gathered, new facts may come to light that invalidate existing aspects of the architecture. A Requirements Impact Assessment assesses the current architecture requirements and specification to identify changes that should be made and the implications of those changes.

See : **32.2.18 Requirements Impact Assessment**

Purpose

Throughout the ADM, new information is collected relating to an architecture. As this information is gathered, new facts may come to light that invalidate existing aspects of the architecture. A Requirements Impact Assessment assesses the current architecture requirements and specification to identify changes that should be made and the implications of those changes.

Content

Typical contents of a Requirements Impact Assessment are :

■ Reference to specific requirements

- Stakeholder priority of the requirements to date
- Phases to be revisited
- Phase to lead on requirements prioritization
- Results of phase investigations and revised priorities
- Recommendations on management of requirements
- Repository reference number


305  Objectives of Requirement Management Phase include
A. Assessing the performance of the architecture and making recommendations for change
B. Ensuring that the Requirement Management Process is sustained and operates for all relevant ADM Phases
C. Managing architecture requirements identified during any execution of the ADM cycle or a Phase
D. A and C above
E. B and C above

Answer : E

Explanation :

The objectives of the Requirements Management Phase are to :

· Ensure that the Requirements Management process is sustained and operates for all relevant ADM phases

· Manage architecture requirements identified during any execution of the ADM cycle or a phase

· Ensure that relevant architecture requirements are available for use by each phase as the phase is executed

306 Complete the sentence.  The Requirements Management Phase _____.

A. addresses and resolves requirements between ADM phases
B. is a central process that prioritizes requirements for some ADM phases
C. is used to dispose of resolved requirements for all ADM phases
D. generates new requirements and passes them to all ADM phases
E. stores requirements and manages their flow into relevant ADM phases

Answer : E

Explanation :

Requirements Management Phase stores requirements and manages their flow into relevant ADM phases.

Requirements Management Phase
- **does not** address and resolve requirements; that is done by the EA after every phase
- is a central process but **does not** prioritize requirements for some ADM phases; thought it is the central process, the prioritisation is done by EA before the requirement reaches this central phase after every other phase

- is **not** used to dispose of resolved requirements for all ADM phases; disposal is done only on realization usually just before phase G

- **does not** generate requirements at all; Business Architecture generates the requirements for the first time and reviews with stakeholder may add to the requirement list, if such requests are accepted as per Change Management norm

Better to read the whole section at this time

16.5.1 General

As indicated by the "Requirements Management" circle at the centre of the ADM graphic, the **ADM is continuously driven by the Requirements Management process**.

It is important to note that the Requirements Management circle denotes not a static set of requirements, but **a dynamic process** whereby requirements for Enterprise Architecture and subsequent changes to those requirements are identified, stored, and fed into and out of the relevant ADM phases, and also between cycles of the ADM.

The ability to deal with changes in requirements is crucial. Architecture is an activity that by its very nature deals with uncertainty and change — the "grey area" between what stakeholders aspire to and what can be specified and engineered as a solution. **Architecture requirements are therefore invariably subject to change in practice.**

Moreover, architecture often deals with drivers and constraints, many of which by their very nature are beyond the control of the enterprise (changing market conditions, new legislation, etc.), and which **can produce changes in requirements in an unforeseen manner.**

Note also that the Requirements Management process itself does not dispose of, address, or prioritize any requirements; this is done within the relevant phase of the ADM. It is **merely the process for managing requirements throughout the overall ADM**.

It is recommended that an Architecture Requirements Repository is used to record and manage all architecture requirements. Unlike the Architecture Requirements Specification, and the Requirements Impact Assessment, the Architecture Requirements Repository can hold **information from multiple ADM cycles**.


307 Which  of the following statements best describes the Requirements Management process ?

A. It is used to develop requirements for Transition Architectures that deliver business value
B. It is used to dispose of, address and prioritize architecture requirements
C. It is used to establish requirements for Phase A
D. It is used to manage architecture requirements throughout the ADM cycle
E. It is used to review business requirements

Answer : D

Explanation :

Requirement Management is the process that is used to manage architecture requirements throughout the ADM cycle

See : **16.5 : Approach (to Requirement Management Phase)**

Note also that the Requirements Management process itself does not dispose of, address, or prioritize any requirements; this is done within the relevant phase of the ADM. It is merely the process for managing requirements throughout the overall ADM.

308 Complete the sentence. The Requirements Management process is used to_____.
A. develop requirements for Transition Architectures that deliver business value
B. dispose of, address and prioritize architecture requirements
C. organize architecture requirements throughout the ADM cycle
D. review business requirements
E. validate requirements between ADM phases

Answer : C

Explanation :

Organizing the architecture requirements throughout the ADM cycle is the primary purpose of Requirement Management Phase.

See : **16.5 : Approach (to Requirement Management Phase)**

Note also that the Requirements Management process itself does not dispose of, address, or prioritize any requirements; this is done within the relevant phase

Note : This question is a mild variation of the question above.

309 Architectural Concern  will not pertain to

A. Affordability
B. Performance
C. Reliability
D. Distribution
E. Security
F. Evolvability

Answer : A

Explanation :

Affordability is not a concern of Architecture. It is a financial concern**.**

See : **Section : 3.34 Concern**

An interest in a system relevant to one or more of its stakeholders.

Note : Concerns may pertain to any aspect of the system's functioning, development, or operation, including considerations such as performance, reliability, security, distribution, and evolvability and may determine the acceptability of the system.

Refer to TOGAF 9.2 online documentation while starting to prepare for Level 2 Questions

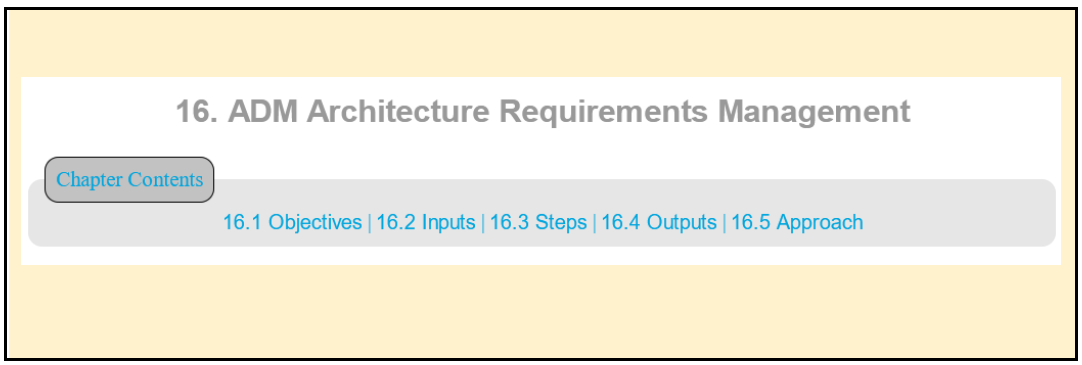
Very important to get to know parts of TOGAF documentation

https://pubs.opengroup.org/architecture/togaf9-doc/arch/

You will need this link to be open most of the time in this course

Similar content will **open during your Level 2 Exam**.

Sorry, not during Level 1 Exam

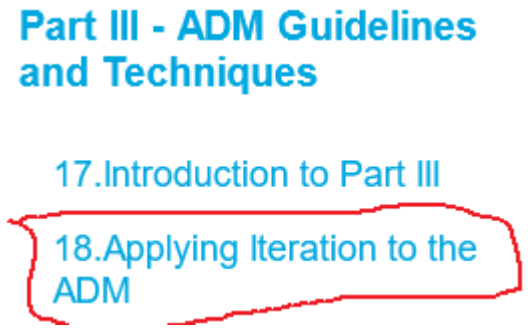


Hint : Though it is termed as a phase in ADM, points to note are :

This is Central Phase, in that it is connected to every Phase of ADM. To this extent, it is a dynamic phase and has relevance to every other Phase.

Questions in Level 2 exam is possible covering the role of Enterprise Architect or the Requirement documentation process performed by an Executive under the directions of the Enterprise Architect.

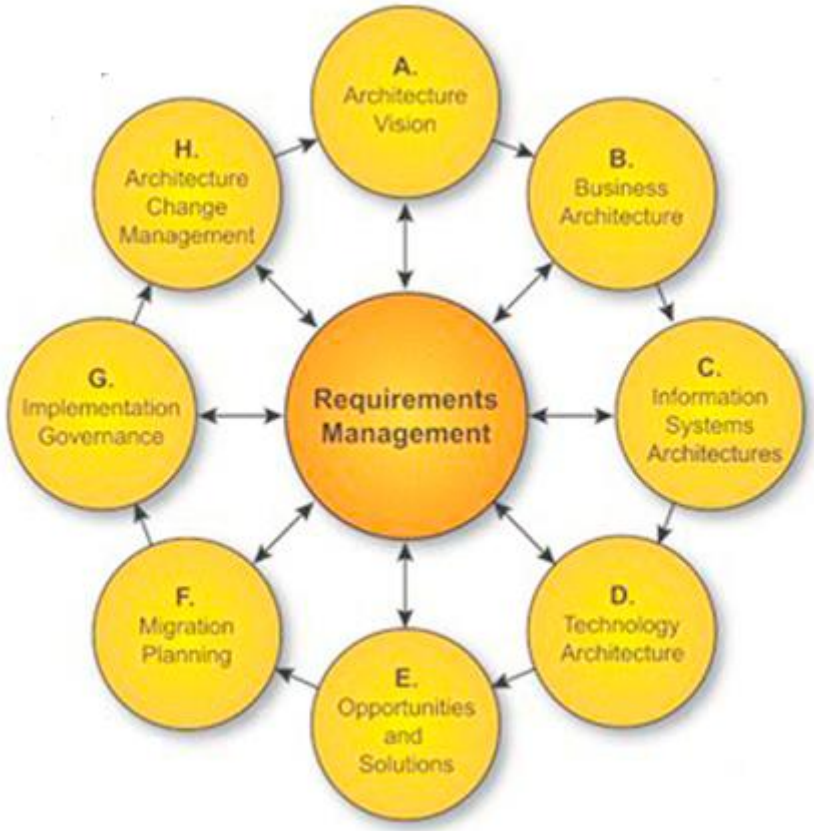
To understand this dynamic phase concept, you may start reading chapter 18, even before we reach it in this course flow :



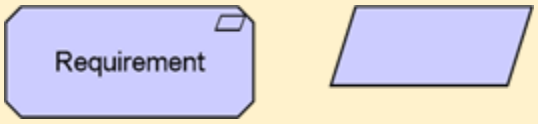
Part 3 : Detailed Courseware

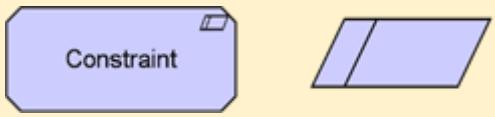
Orbus Software video on this Phase :

https://www.youtube.com/watch?v=W_K-KbTeGtk&t=29s



Pictorial Quiz

How am I 

different from 

?

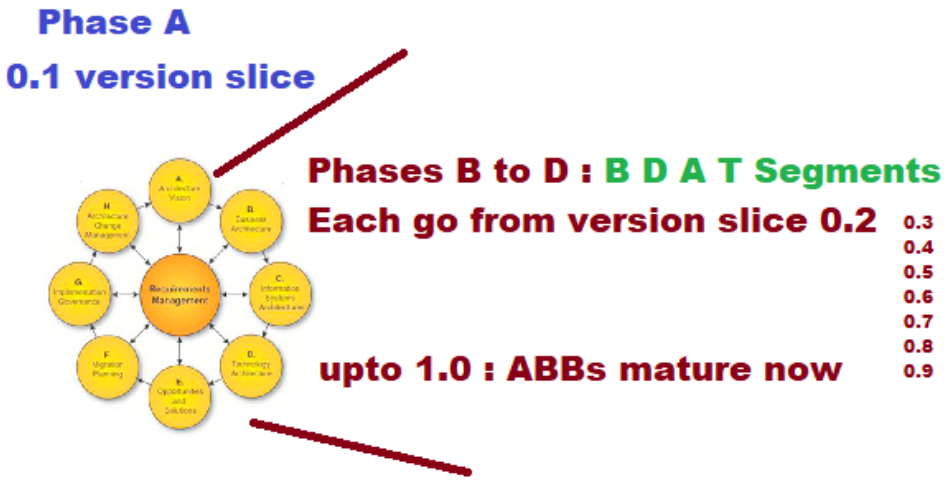
Design and Development simply take requirement into executables.

But Architecture also takes into account the Big Picture,

Including Constraints, Service Qualities and more

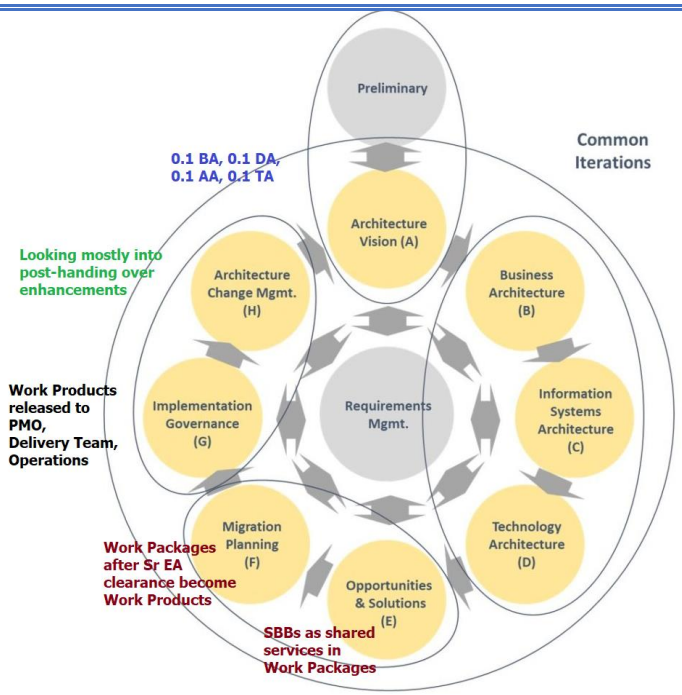
The Happening Story

The EA team plays an important role, while a (probably a non-Technical) Requirement Management Executive plays a document management role, whenever this Phase is referred.



Why slices like 0.2 up to 1.0 ?
To help you split out the project understanding in slices of versions

									Bounded Context, Context Maps			
0	1.0	0.9	0.8	1.0		Outline, sketch, in ADD, ARs						
Week	BA	AA	DA	AT		What's			DDD			



Sample Flow, in a portfolio, shown in part
0.2 BA -> Reviews & Acceptance by EA
0.3 BA -> Reviews & Acceptance by EA

0.2 AA -> Reviews & Acceptance by EA

0.2 DA -> Reviews & Acceptance by EAhis

like this till 0.9 BA and 0.5 AA, 0.5 DA

then
TA.02,...TA 0.3 ...

finally
1.0 BA, 1.0 AA, 1.0 DA,1.0 : Plateau, ABBs mature

The Chief EA or Senior EA who is in charge of a Portfolio of Projects will play the role of accepting Architectural work done by the B D A T Architects. That is done at the end of each version in Phase B or Phase C or Phase D as the case may be.

The role extends to setting priority of work and also that of deciding if BA 0.1 work is accepted,

i) should it go to which of the D A T for 0.1 work or should it again go to BA for more by nature of 0.2 work (**Where next ?, after first installment of BA work, under 0.1 version slice**)

Week	BA	AA	DA	AT		What ?		DDD
1	0.5					WW, Use Case 1 - Order Selection		
	Approved by EA							
3	0.3					WW one more Use Case - Checkout		
4	-	0.5				Aggregates first look		
			0.5			Aggregate event data process		
	Approved by EA							
2	0.4	0.3				BA Payment;		AA More Aggregates
6			0.3			Event Bus early design		
7	0.2			0.5		BA : Order Confirmation; TA basic infra		

ii) When should such a work go to the next in line – the timing based on priority and workload (**When** is the next work, BA 0.3 to be scheduled ?)

Sample Flow, in a portfolio, shown in part
0.2 BA -> Reviews & Acceptance by EA
0.3 BA -> Reviews & Acceptance by EA

0.2 AA -> Reviews & Acceptance by EA

0.2 DA -> Reviews & Acceptance by EAhis

like this till 0.9 BA and 0.5 AA, 0.5 DA

then
TA.02,...TA 0.3 ...

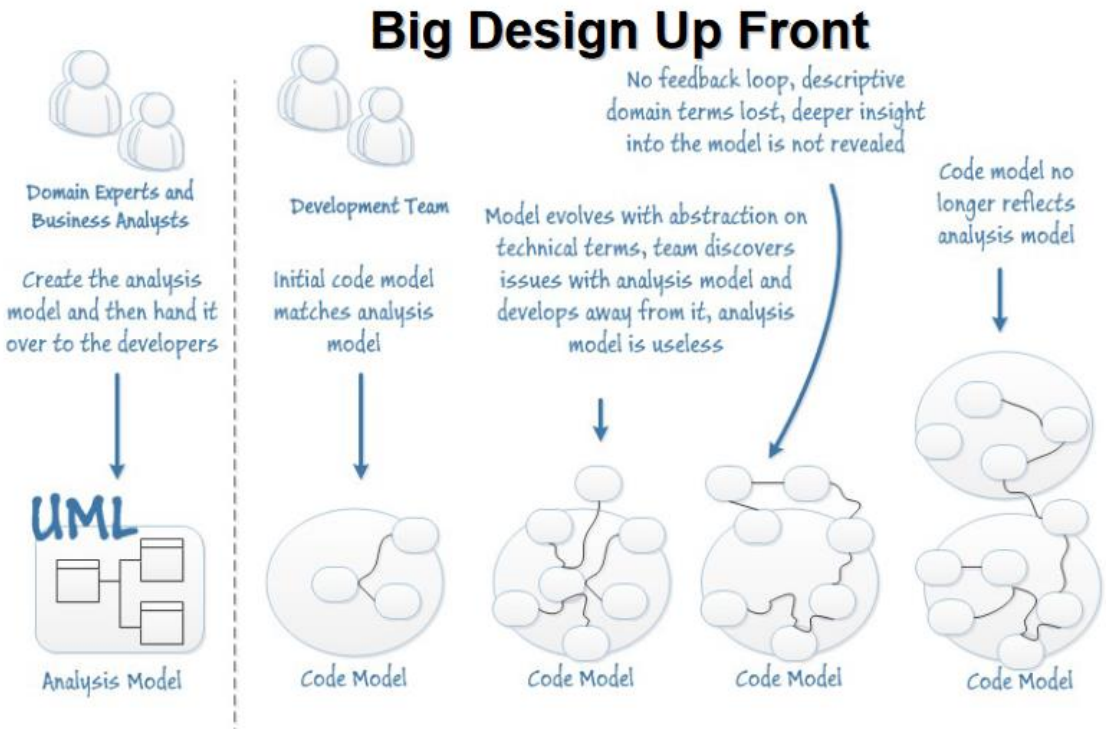
finally
1.0 BA, 1.0 AA, 1.0 DA,1.0 : Plateau, ABBs mature

Week	BA	AA	DA	TA		What ?					
x	1.0	1.0	1.0	1.0		Final review of All ABBs; move to Phase E					
Ph E : Full Stack											
x	Change Attributes; Determine Constraints										
x+1	Consolidate Gap Analysis; consolidated requirements										
x+2	Reconcile Interoperability ; Validate Dependencies										
x+3	Readiness and Risk Assessment; Work Packages ; Transition RoadMap										

The Requirement Management Executive will **track and store** this status and will ensure that the work is forwarded to the right person just before the right time. Actually this task can be automated by a dashboard style scheduler software also.

Nice to Know Box

What is wrong in proceeding without any version slices during the 'Design Phases' ?



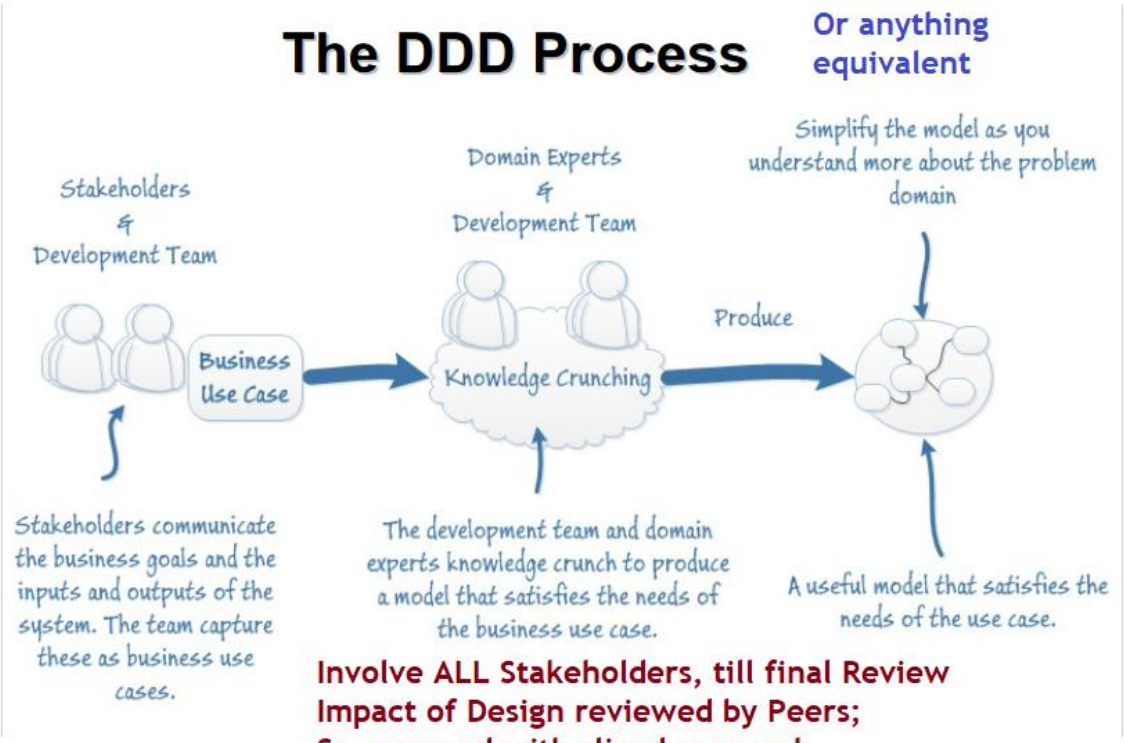
Big Design Up Front = older approach

No Slices during Design Process

Aimed at humongous Monolithic

Nice to Know Box

Hence, TOGAF advocates :



Involve ALL Stakeholders, till final Review
Impact of Design reviewed by Peers;
So, proceed with sliced approach

Nice to Know Box

These version slices are meant for Architects only, purely for the purpose of '**managing the requirement'** as it turns from captured Requirement to Architectured ABBs.

This should not be mistaken against the following which are Development time version naming meant for Release Flow :

Semantic Versioning

`<major>.<minor>.<patch>[.<build number>]` (1.20.11, 1.20.11.403)

Calendar Versioning

`<year>.<month>.<patch>` (2022.07.01)

Sprint-based Versioning

`<symbol><sprint number>[.<patch>]` (M27, M27.1)

Nice to Know Box

Why not a single step Design and single step Execution ?

Do not attempt to do everything at once.

Every large-scale initiatives must be accomplished iteratively.

It is easier to maintain a sharper focus on each effort, by organizing work into smaller, manageable sections that can be executed and completed in a timely manner.

While firming up the Requirement, in Phase B to D – Kickstarted by Phase A :

Make ABBs – Architecture Building Blocks ; that too as small and incremental version slices : Version 0.1, 0.2 etc.,

That too for each segment :

BA : 0.1, …. 0.2 and so on

AA : 0.1, …. 0.2 and so on

DA : 0.1, …. 0.2 and so on

TA : 0.1, …. 0.2 and so on

Nice to Know Box

While designing Platform Specific Solution Building Blocks in Phase E :

Combined Full Stack (Business + Data + Application + Technology Infrastructure) Solution Building Blocks as Transition Architectures that suits Agile / Dev Ops / CD : Continuous Development – CI : Continuous Integration (Continuous Deployment through suitable Transition stages)

All the above version slices and Transition Architectures are based on feedback before, throughout, and after each small iteration so that it will ensure that actions are focused and appropriate, even if circumstances change.

Nice to Know Box

- Ensure that relevant Architecture Requirements are available for use by each Phase as and when it is to be executed

Push Requirements, that are in some version number (say 0.1 to 0.9) to the appropriate Phase which will act next on it, based on Priority

Nice to Know Box

In concrete terms, a set of requirements determines what must actually be implemented, and conversely, what is not retained.

Based on given business goals, concrete requirements generally translate to how different factors, be they technical, budgetary or organizational, are to be taken into account.

Quiz Time

If I am the Requirement Document Management Executive, what is my work under your leadership ?

Nice to Know Box

ADM is **continuously driven** by the requirements management process.

It is important to note that the Requirements Management 

circle denotes, not a static set of requirements,

but a dynamic (forever) process

Quiz Time

Requirements are dynamic because

Change Requests that are accepted during project : Yes / No

Changes would come during the project and thereafter as additional requirements for next release. So it is almost continuous

Additions over existing projects : Yes / No, as Brownfield

Both

Fresh Greenfield requests: Yes, from Phase A / No, in any Phase

Fresh Greenfield projects must start from Phase A

Nice to Know Box

Baseline : Where we are :

Do not start from scratch and build something new without considering what is already available.

There is likely to be a great deal in the current services, processes, programmes, projects, and people that can be used to create the desired outcome. Investigate and observe the current situation to ensure that it is fully understood.

Nice to Know Box

This is not Product Release version number like Windows 10 and Windows 11. That is more of a version based on Product feature lists. Often the Transition Architecture (like T1, T2, T3 etc.,) as laid out in Phase E may be synchronized with such a Feature Release list.

Then why a version number mechanism, that too only between Phases A to D ?

This is a number indicating progress of Architectural Design only, which starts with outline in Phase A but gets complete Architectural description, Architectural structure and so on as a Platform independent Architecture.

Every project or portfolio need to have version numbers 0.1 to 1.0 but every project / portfolio **need not have ten version slices** as discussed in this connection. TOGAF just suggests ten slices as a sample.

For example, a portfolio may have

0.1 : Outline as appearing in Phase A

0.5 : All collaborative needs between users / project modules are captured and structured as ABBs

0.7 : All security focus points on the above are added to the ABBs

1.0 All other requirement aspects including generic scalability approaches, user Experience facilitations, modules in various Tiers and Layers as an abstract High Level to Medium Level Design is complete. The BA / DA / AA / TA portions are sufficiently reviewed and deemed to have reached a plateau level.

Such a version slice planning is done in vision Phase. this is because it is 'slice' of the vision into suitable parts so that Phases B to D can take it up in suitable installment approaches. Take is as a Vision breakdown for convenience of High to Medium level design work.

Sometimes, the same version slices may coincide with the Product Release versioning plan. It need not be so always.

Quiz Time

Version number legal / illegal as per TOGAF :

0.0 : Yes / No :

No, it starts with 0.1

Vision Phase outputs the first versions

These are :

- Baseline Business Architecture, Version 0.1
- Baseline Technology Architecture, Version 0.1
- Baseline Data Architecture, Version 0.1
- Baseline Application Architecture, Version 0.1
- Target Business Architecture, Version 0.1
- Target Technology Architecture, Version 0.1
- Target Data Architecture, Version 0.1
- Target Application Architecture, Version 0.1

All above are placed duly pointed to by ADD

2.0 : Yes / No

No. It stops at 1.0

1.15 : Yes / No

No. The last number is 1.0

0.1 0.2 …… 0.9 1.0 : Yes / No

Yes. Quite legal

0.1 0.15 0.2 0.25 …. 0.9 0.95 1.0 : Yes / No

Yes. Any sub-numbering between 0.1 and 1.0 is fine

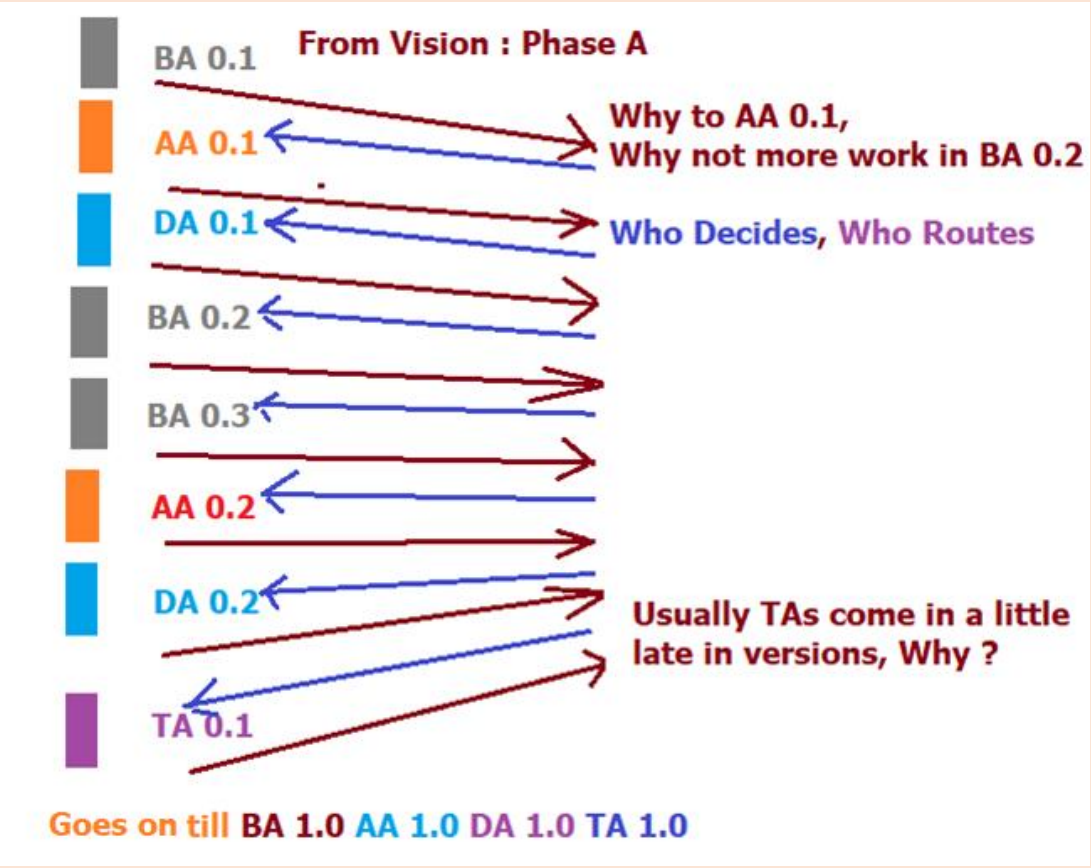
This is same as Release numbers : Yes / No

No. Software or hardware release numbers are on completion

Completion in the form of being released to the user

These are versions numbers to indicate Architectural progress

Nice to Know Box



Who decides ? EA : Usually Sr EA or EA

Who Executes ? : Segment Architect. BA / AA / DA / TA

Who Manages the Routing ? : Requirement Management Executive

Quiz Time

Version number has to be :

BA 0.2 always followed by AA 0.2: Yes / No

No. the EA may decide that
BA 0.2 is followed by BA 0.3
Or followed by AA 0.2 or DA 0.2
Very rarely it is immediately followed by TA 0.2

TA 0.1 may start much later than BA 0.1, AA 0.1, DA 0.1 : Yes / No

Yes. Very often, Yes

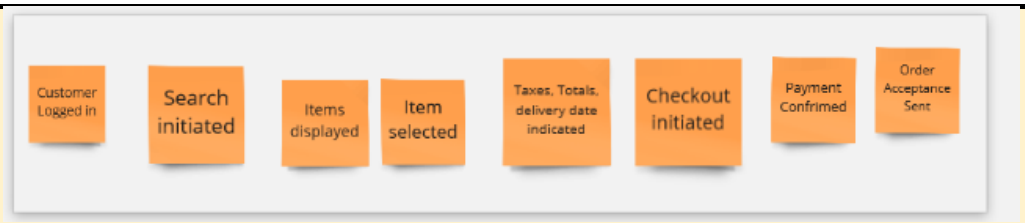
Relating to the **Case Study**

Nice to Know Box

Points of Essence : The Path of Requirements moving forward

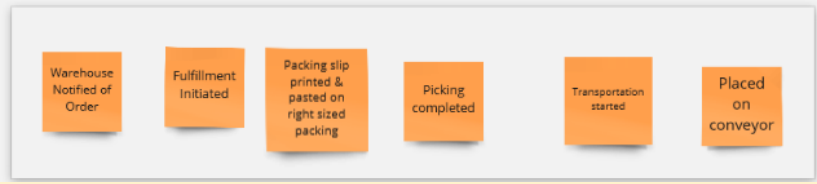
Documented : Ten Architectural Building Block version slices in this Portfolio, in the journey Phase A to Phase D

Reminded relative Priority to all involved. Requirements firmed up, with Target of one slice taken as Baseline of next. DevOps Team got involved early enough. Event Storming technique was used towards understanding the Requirements by the 'Team'.

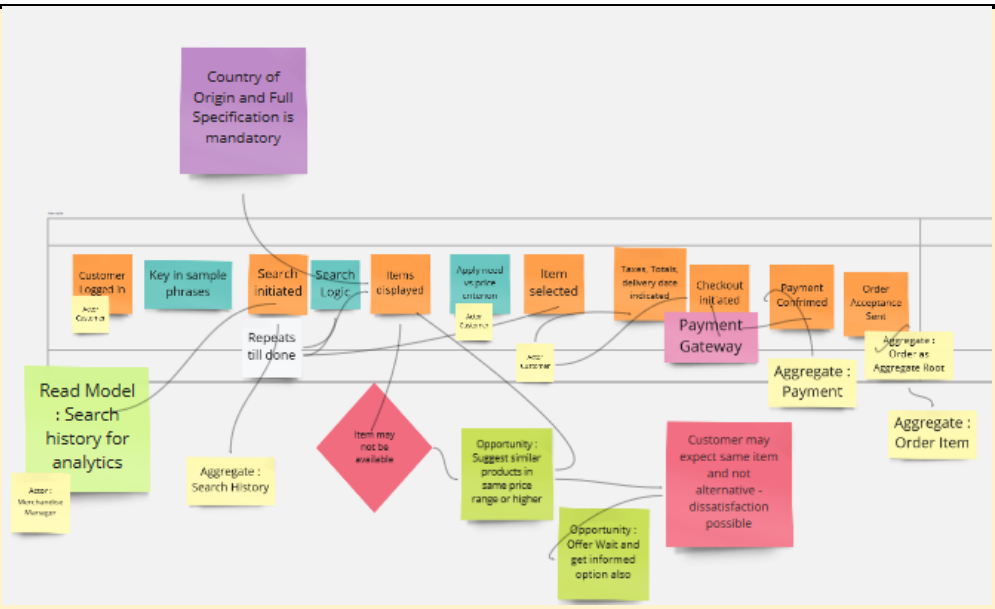


This could have been any other Requirement Capture Technique. Any Technique including some kind of Visual Domain Story Telling, Story Boarding, User story or any such incremental design approach. In effect it should be an approach which starts from Business analysis and remains in close touch with all other Architectural steps and perhaps with a tinge of Agile Story Mapping behind it. We need to make sure that no-one in the Team or in the Stakeholder list is left behind in this journey.

We just used Event Storming, to be in line with DDD – Domain Driven Design (Eric Evans) and Microservices which are 'scoped' in Phase A itself.



The Architectural Building Blocks, the ones which gradually converted the Stakeholder concerns into Requirements and then into Platform Independent Model evolved in our case through Event Storming and DDD.



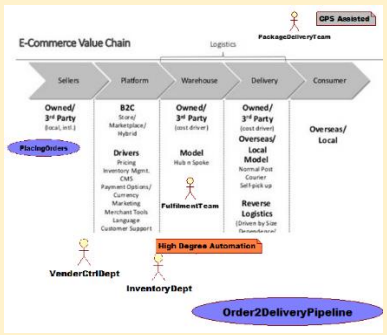
The ABBs got firmed up quite a bit by the time the Architecture work reached 1.0 version slice (starting with version 01. Which is just the Aspirational Vision)

Amazon : Their vision is "To be the world's most customer-centric company."

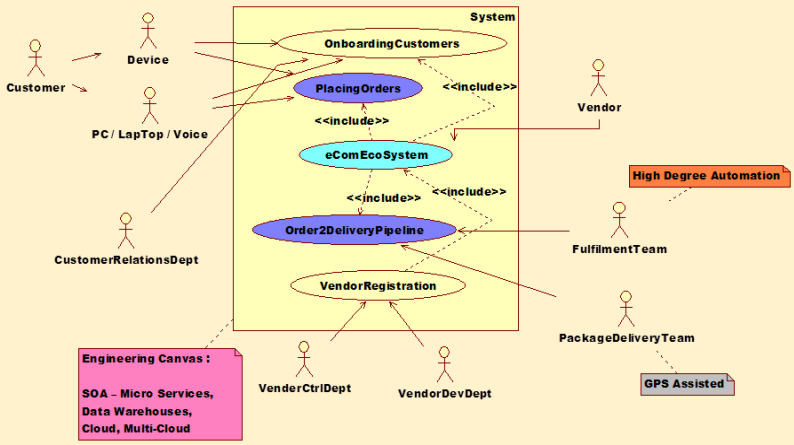
Wikipedia : Their vision is "A world in which every single person is given free access to the sum of all human knowledge."

Harvard University : Their vision is "To develop leaders who will one day make a global difference."

Requirement Management tracked a few Artifact Building Blocks initiated in Phase A :



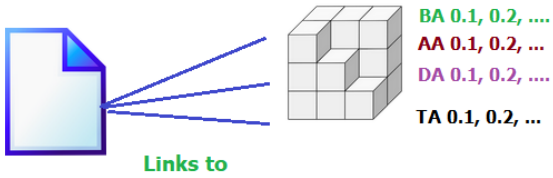
Value Chain, duly related to Solution Concept Diagram



Conceptual Business Scenario

Architecture Definition Document :

Deliverable container
for the core architectural artifacts



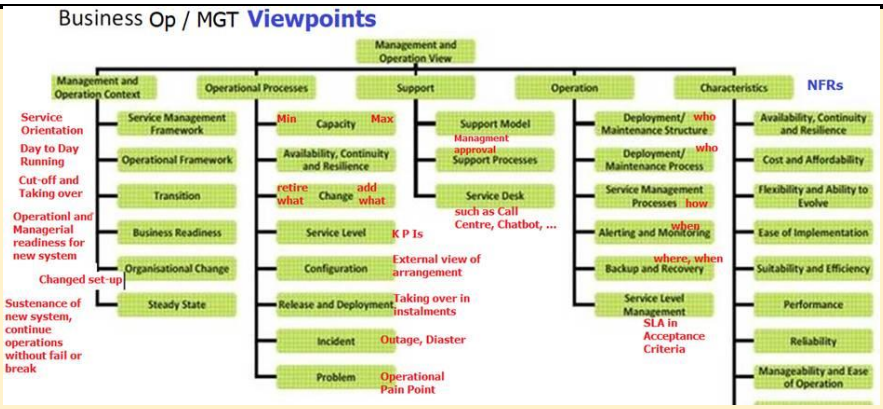
Architecture Requirements Specification

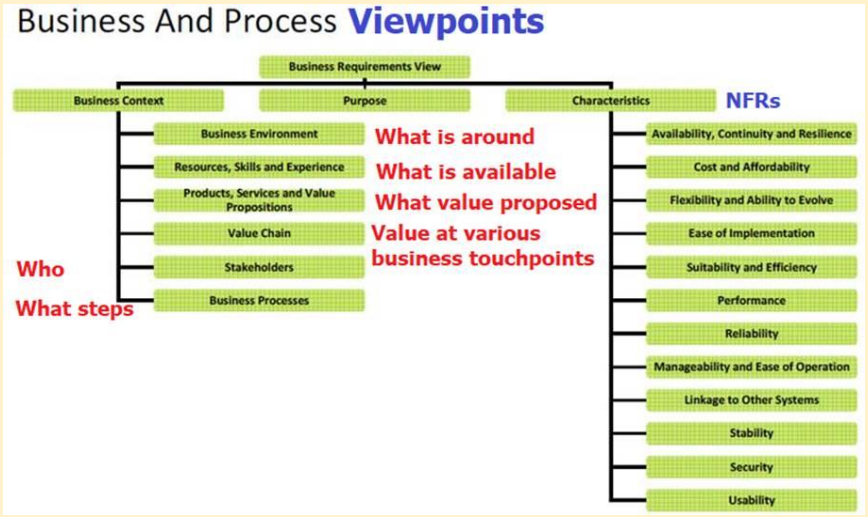
Contains sections like :

2.1	**Architecture Requirements**
2.2	**Interoperability Requirements**
2.3	**Constraints**
2.4	**Assumptions**
2.5	**Success Measures**

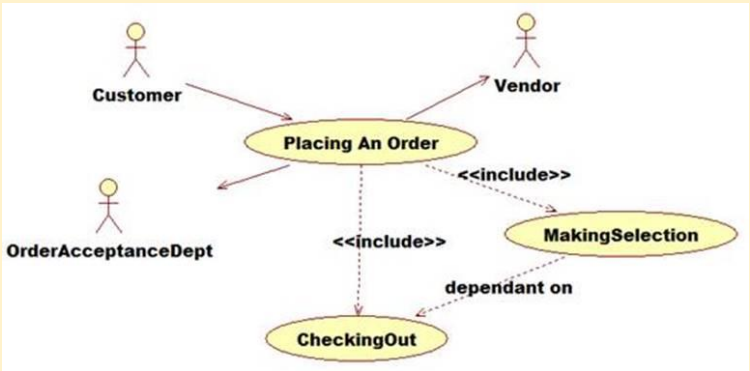
The journey from version slice 0.2 to 1.0 has its own taste of :

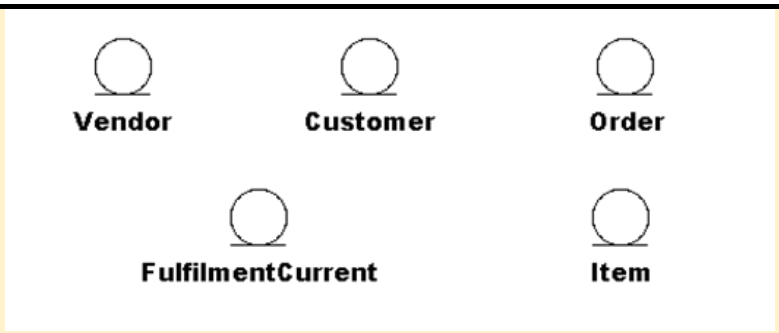
Business Architects : Strategic to Operational thinking

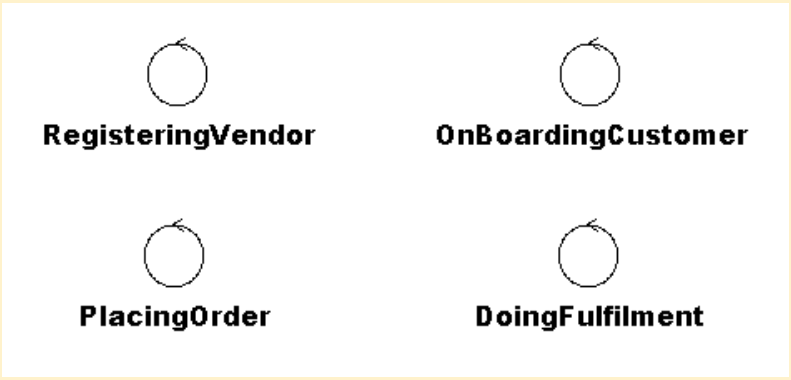


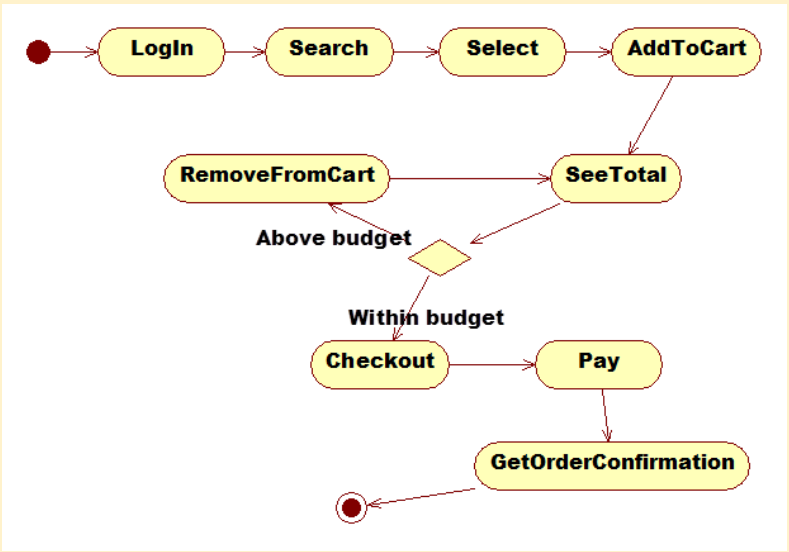


They went in line with Business Scenario technique as advocated by TOGAF. They added Use Cases, Activity Model, Robustness (MVC) Class Models and gave a clearer picture to the needs of the Portfolio which is in tune with the Goals and Drivers.



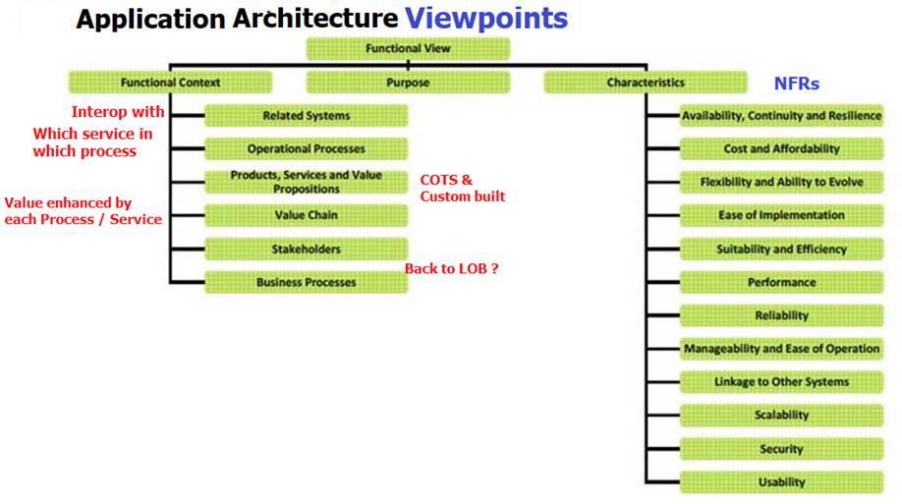


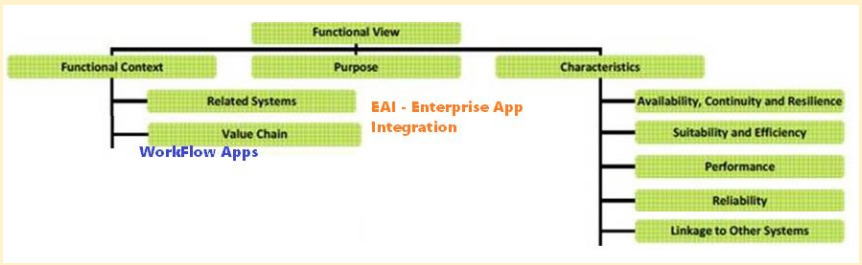




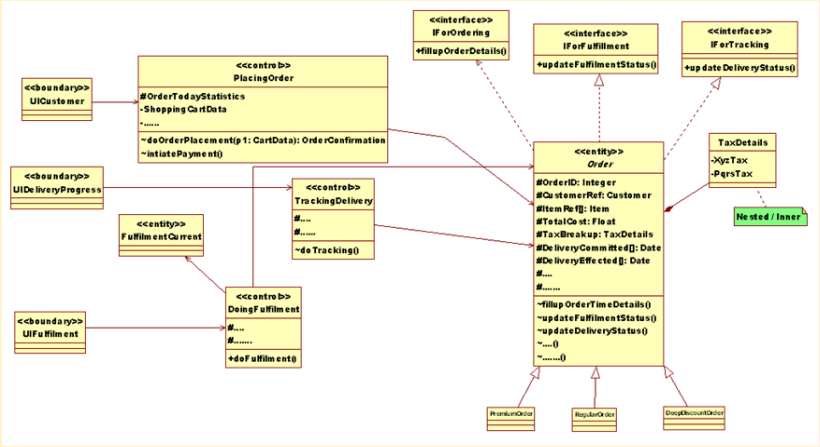
Application Architects : Shaping the core components (yet not attaching to specific implementation product but focusing on independent services that could spring out from the business and stakeholder needs). Focused on

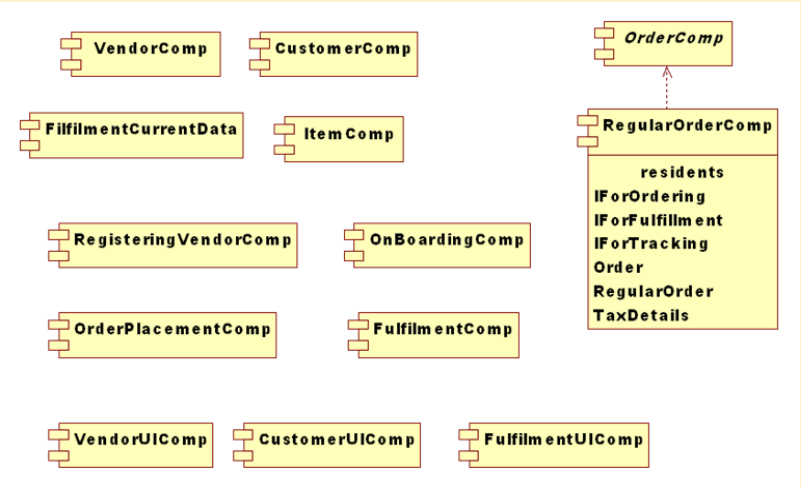
reusability of common services where it makes sense.

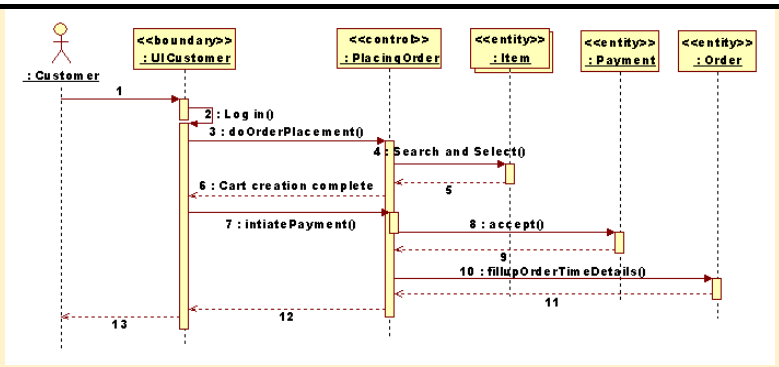


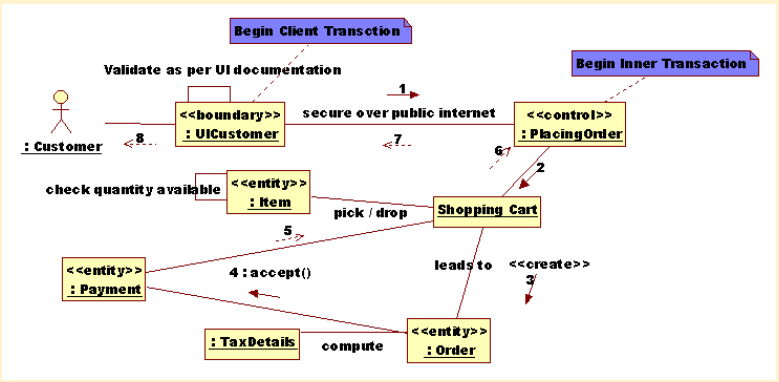


- Application/Organization matrix
- Role/Application matrix
- Application Interaction matrix
- Application/Function matrix

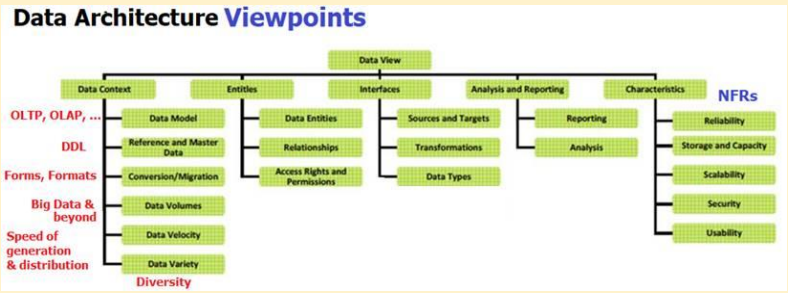








Data Architects : Closely working with Application Architects : In data store design (and everything in Platform Independent manner) : Stores factored out of SQL-RDBMS, NoSQL and other modern and Cloud biased variants, Warehouses and Marts and Lakes, enabling direct queries to complex BI – business Intelligence and Data visualization – with Analytics playing its due role. Event Driven approach taken suitably since each Microservice may have its own data store and gets connected to an Event queue in some kind of persistent message store.



Information/Data

- Data Integration
- Data Architecture
- Master Data Mgmt
- Metadata Mgmt
- Data Delivery Architecture
- Dashboards & Analytics
- Business Intelligence
- Enterprise Reporting
- Corporate Performance Mgmt
- Data Modeling
- Data Quality
- Content Mgmt

Management, Migration and Governance seen at micro levels is to be sensed over these points

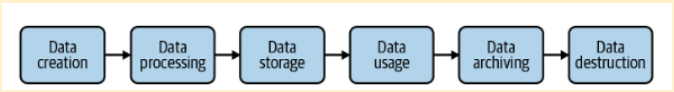
Data Visualization

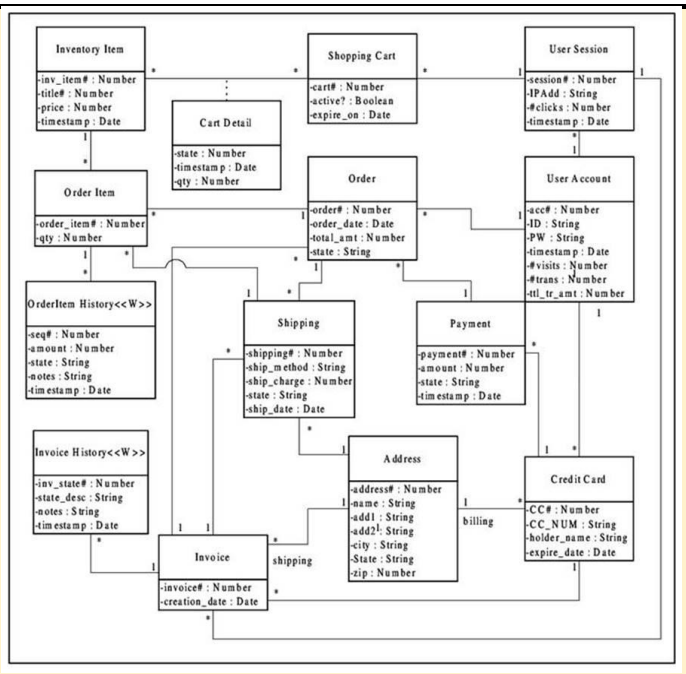
Modeling is about ABB, SBB design

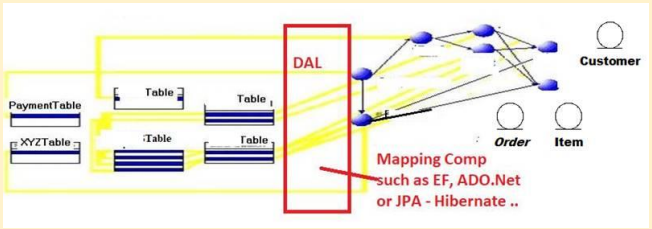
Content Mgmt covers Repository, EC

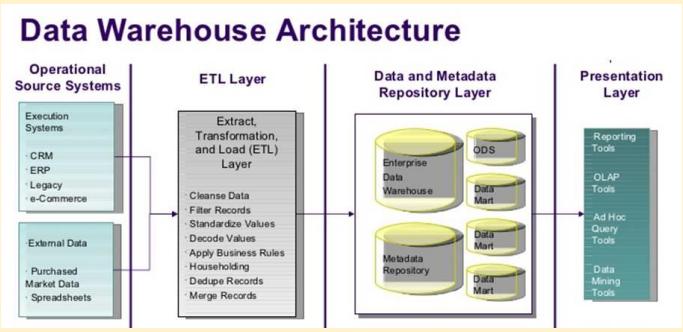
- Data Entity/Data Component catalog

- Data Entity/Business Function matrix
- Business Service/Information matrix
- Application/Data matrix

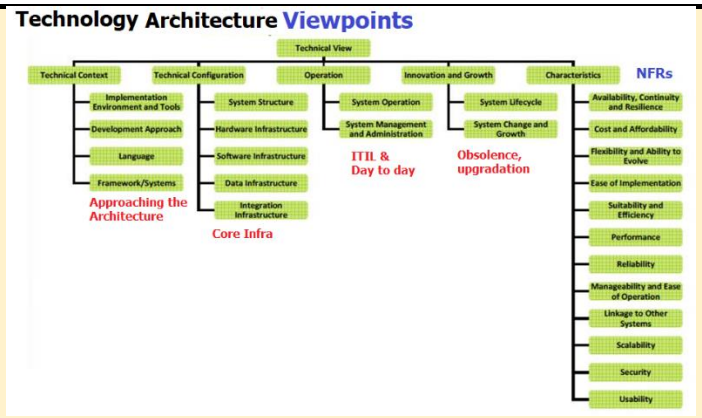








Technology Infrastructure Architects : Stepped in at the right stage of the Portfolios. Looked into all hosting Operational Platform, hosting Hardware systems, Cloud subscriptions and connection management, network management and so on. Since NFRs and Security features as duly architectured by others gets into a platform culmination at this, more attention is given hereon.



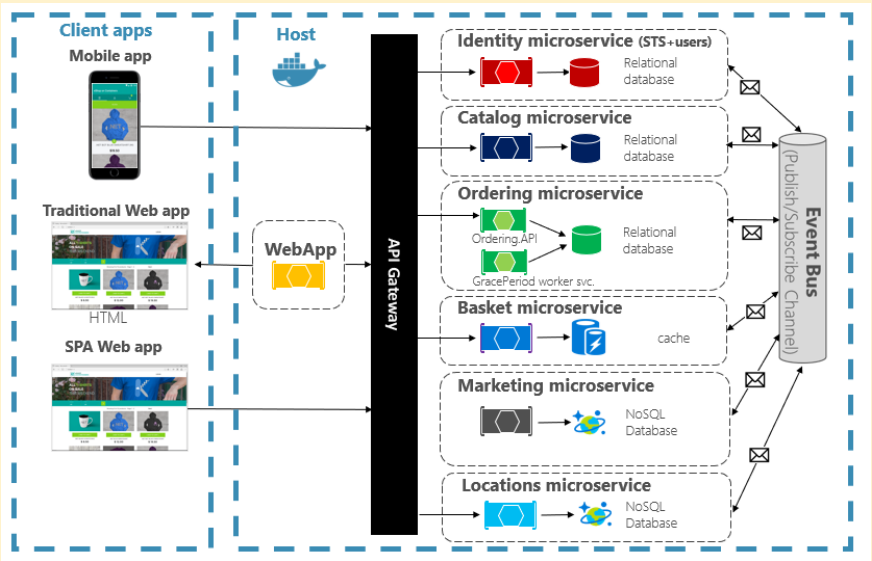
More of Enterprise wide nature

- Technology standards catalog
- Technology portfolio catalog

More of Project Specific nature

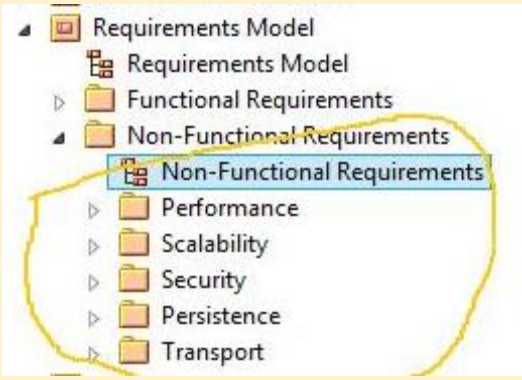
- Application/Technology matrix

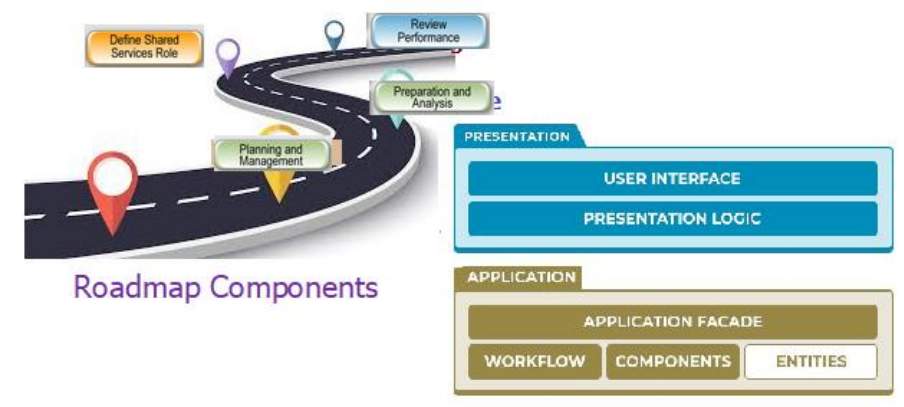
- Environments and Locations diagram
- Platform Decomposition diagram
- Processing diagram
- Networked Computing/Hardware diagram
- Network and Communications diagram

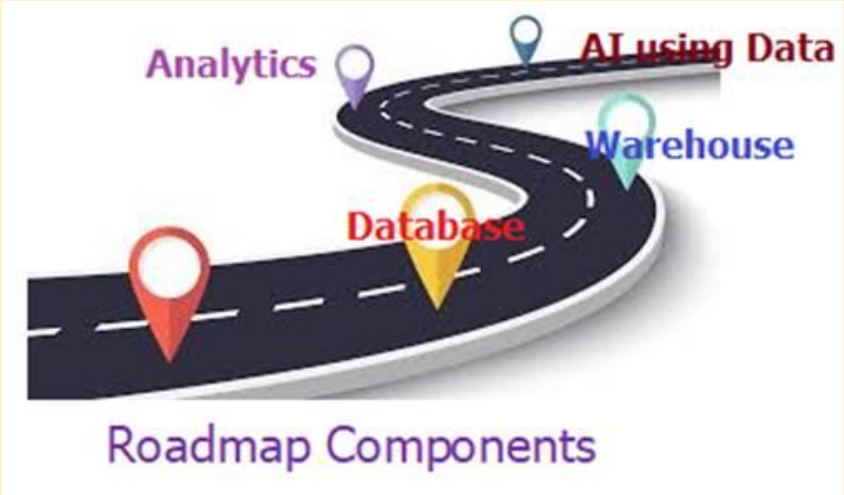


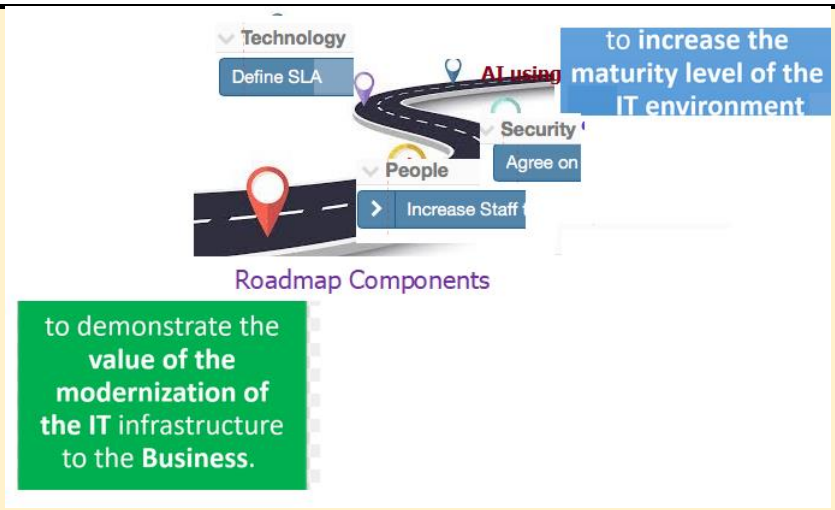
Phases E and F, which are highly service centric and with Platform (Specific) Dependent Design proceeds at this stage. Here onwards the Requirement Management tracks Transition Architectures with SBBs in them, work Packages, Work Products, oversight over Development and Installation by PMO and oversight by Operations Department for CD / CI and so on.

At this stage, the broader Roadmap needs a closer look :


Requirements Model
Requirements Model
Functional Requirements
Non-Functional Requirements
Non-Functional Requirements
Performance
Scalability
Security
Persistence
Transport


Define Shared Services Role
Review Performance
Preparation and Analysis
Planning and Management
Roadmap Components
PRESENTATION
USER INTERFACE
PRESENTATION LOGIC
APPLICATION
APPLICATION FACADE
WORKFLOW
COMPONENTS
ENTITIES


Analytics
AI using Data
Warehouse
Database
Roadmap Components



Nice to Know Box

Document the Requirements - use business scenarios, or an analogous technique : Note that capturing the Requirement by engaging with eh Stakeholders is not done by this Phase. It is part of steps in other Phases

Identify the Requirement Documents, as generated by BA

Means, recognize the version number of the Architectural Progress

Nice to Know Box

Baseline – what work done till now – Available at any point of work from Requirements Repository

Look into the Baseline to Determine and Record the Priorities : Storing them in Requirement Repository

WIP : Work In Progress

Nice to Know Box

Monitor baseline requirements

Continue the process of monitoring the Baseline Requirements so that fresh Targets are achieved and they become Baseline

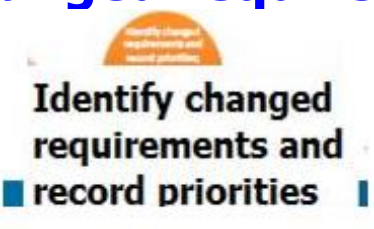
Baseline version number demands move to a Target version number. Monitor this

Note that **priority is decided by Enterprise Architect**

Document Executive simply makes a note so that action, when needed, can be promoted

Nice to Know Box

Identify changed requirements and record priorities :



Be alive to change in Requirements. Would need change in priorities as well as change in documentation

This is Document Executive noting the priority after an Architect completes work of a version (say 0.1 to 0.2) in the Project

Nice to Know Box

Based on the priority, some work (Gap Analysis and producing Architectural Building Blocks) is done in every version slice. This is done with due engagement of Stakeholders

Do record the decisions related to the requirements as and when the Gap Analysis and Building Block preparation work is performed in respective Phases

WIP : Work In Progress

Nice to Know Box

Assess and revise gap analysis for past Phases

The gap analysis in the ADM Phases B through D identifies the gaps between Baseline and Target Architectures. Certain types of gap can give rise to gap requirements.

EA always looks into this while fixing Baseline and Target each time

EA looks at current status and state of Architecture before setting more Targets. Done many times in B, C, D based on completed and accepted version number.

.

Nice to Know Box

Enterprise Architects look into Impact Analysis as produced by down the line Segment Architects in Phases B to D

Impact Analysis done as steps in Phases B, C, D :

Assess impact of changed requirements

Record the result of assessment relating to impact of every Building Block in Requirements Impact Statement

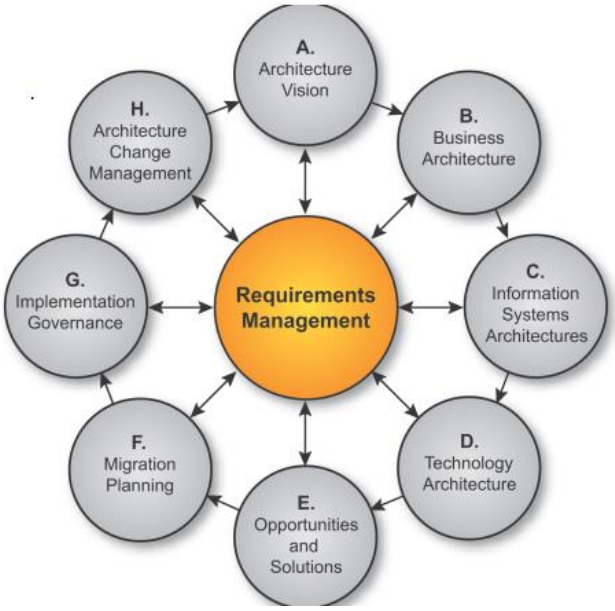
a. Assess impact of changed requirements on current (active) Phase

b. Assess impact of changed requirements on previous Phases

Solution Architects conduct an Impact review (across Architectural Landscape) at various stages of their work. These need to be assessed by EA and further course of action of routing depends on its contents

Assessment by EA, Segment Architects,

leading to Requirements Impact Statement



c. Determine whether to implement change, or defer to later ADM cycle; if decision is to implement, assess timescale for change management implementation

Decision by EA

d. Issue Requirements Impact Statement, Version *n*+1

Requirements Impact Assessment

Physical report managed by Document Executive

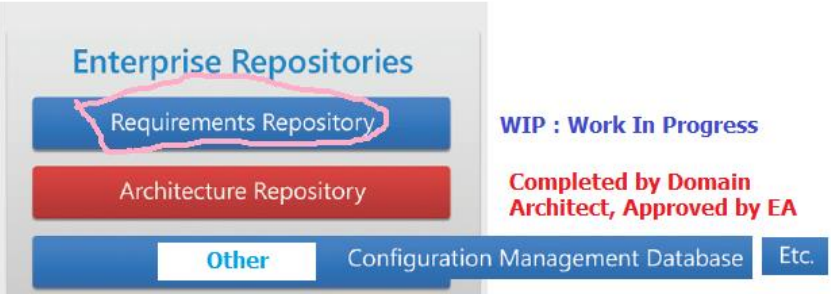
Purpose

Throughout the ADM, new information is collected relating to an Architecture. As this information is gathered, new facts may come to light that invalidate existing aspects of the Architecture. A Requirements Impact Assessment assesses the current Architecture Requirements and specification to identify changes that should be made and the implications of those changes.

Content

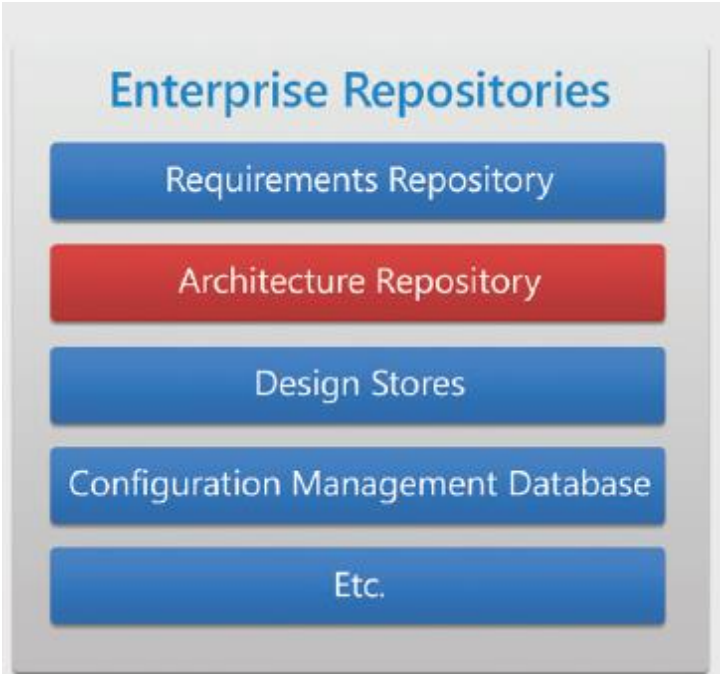
Typical contents of a Requirements Impact Assessment are :

- Reference to specific requirements
- Stakeholder priority of the requirements to date
- Phases to be revisited
- Phase to lead on requirements prioritization
- Results of Phase investigations and revised priorities
- Recommendations on management of requirements
- Repository reference number



Requirements Repository – Made accessible to selected EA Team members

Architecture Repository – Accessible to all connected to Architecture and projects


Enterprise Repositories
Requirements Repository
Architecture Repository
Design Stores
Configuration Management Database
Etc.

Nice to Know Box

What to Expect in a Well-Run EA Repository : Architecture Requirements Repository

Managing requirements to the entire EA Landscape is one of the most complex activities facing the Practitioner. The first challenge is simply the breadth of detail; the second challenge is the overlapping nature of managing requirements across the EA; the third challenge is maintaining the repository over time; and potentially the fourth is integrating with other repositories.

One thing that is important to consider is that requirements appear radically different depending upon the purpose of the architecture and the level of detail. As an extreme example, Practitioners with experience in solution delivery architecture and implementation **may not recognize** requirements for architecture developed to support strategy as requirements. Practitioners used to implementation tend to be **looking for very granular requirements** to express statements of need. Be agile, be efficient, integrate the new division, and protect the market-leading differentiators are all examples of key requirements for Architecture that supports Strategy and Portfolio.

Leading practices find that a large number of requirements for Architecture that supports Portfolio and Project are normally captured in the form of scores. Ask the stakeholders to assess the required efficiency, maturity, automation of a process, application, service or capability; score the required business fit or technical fit of applications; and score the preferred lifespan of the infrastructure. Best practice is to use a scale of one to five to capture their assessments. All of these scores are requirements; they clearly state the preferences of the stakeholders.

An important question in any Requirements Repository is whether these are architectural requirements or implementation requirements. The distinction can be fine, but it is a distinction with a very large difference.

One of the tests that can be used for distinguishing between architecture and implementation design is whether the description can only be done one way, or can it be realized multiple ways.

The former tends to be architecture, while the latter is implementation design. **When an Architecture Repository is integrated with a Requirements Repository for implementation, use appropriate integration options to maintain traceability and integrity.**

Many architecture requirements are remarkably long-lived. Especially when the requirement is articulating aspects of the Enterprise that differentiate it.

When does a market leader who leads through customer experience want to relax the requirement requesting best-in-class customer experience ? The real challenge for the Practitioner is translating market-leading customer experience into clear architecture specifications applied to components in the architecture.

Herein lies one of the mental challenges when architecting for different purposes – the **line between a requirement and a specification may be in who stated it.** A requirement into a portfolio architecture aimed at market-leading customer experience may result in an architecture specification requiring that the information object "customer preference" be a common information object to the CRM, customer portal, and service desk. That specification reads like a requirement to the architect supporting solution delivery of the new CRM.

Requirements from higher in the organization also tend to be discussed using different names. It is common to speak of objectives and mandates, and treat them with special reverence.

Likewise, the distinction between types of requirement – functional versus non-functional, business requirements versus technical requirements – is treated very seriously. In the final analysis, whether a requirement is a mandate, a non-functional requirement, or a business requirement, from the perspective of a Practitioner it is a statement of need that will be addressed in the context of the superior architecture and the set of objectives provided by all stakeholders.

One central activity Practitioners typically are not comfortable doing is assessing the validity of requirements. When the Practitioner has a well-described strategy, a portfolio that identifies gaps, and gap-filling work packages, it becomes easy to look at a requirement being injected in the project or solution delivery architecture and assess whether this requirement is in conformance with what the Enterprise priority is or whether this requirement conflicts with the superior architecture.

Consider a portfolio initiative focused on improving agility for customer experience: this portfolio will identify a set of projects explicitly designed to improve some aspect of the customer experience and improve the ability of the Enterprise to change. As time progresses close to execution, it is common for requirements not aligned with the project's purpose to be injected into the process.

The central element of requirements management is good governance. Practitioners are guardians of the statements of value.

When Practitioners have a good architecture identifying the target and transition steps along the way, requirements, and architecture specifications, may vary over time; be different in the target and the transition architectures. Imagine a portfolio roadmap that deliberately sacrifices customer experience for agility in the first transition. Then in the second transition the priority switches and agility are sacrificed for customer experience. The conformance test to architecture requirement, and guidance on priority, switches.

When clarity of precedence and priority is not available, data to guide trade-off early in the cycle is absent, hindering progress. Just as the assessment of precedence and priority shifts context to other decisions where a set of preferences are well defined and is closer to the organization most suited to make the choice.

Explicitly link the architecture specification to requirements, and trace the requirements to a stakeholder / concern pair to track the value and preference. This traceability is used in governance to assess how well the design and implementation choices address the stakeholder's value preferences.

Best practice **EA Repositories facilitate traceability at every step of the architecture** to the **direction and priorities of the Enterprise**. Practitioners are delivering some of the highest value when they are engaged in Requirements Management and trade-off. All smart stakeholders want all, want more, and for free. All smart stakeholders know they cannot have it all, nor can they have it for free. What stakeholders do not know – and what the role of the Practitioner is – is to assist the stakeholders in understanding what they have to give up in order to realize different sets of preferences.

A Practitioner with a well-run EA Repository is in a position **to maintain a comprehensive set of requirements in context.** Requirements in context enable the Practitioner to work actively for the preferences of the stakeholders rather than architecting to a subset of the preferences of the stakeholders; or worse a set of preferences that the Practitioner personally prefers.

(From TOGAF 10 documentation, Relevant for TOGAF 9.2 also)

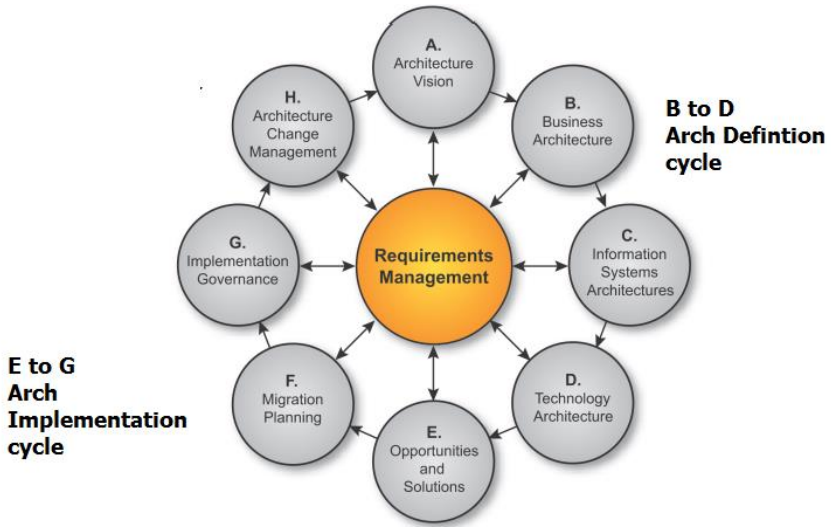
Nice to Know Box

Implement change in the current Phase

Implement change in the current Phase : Means, role of Requirement Management is to keep track of such a progress

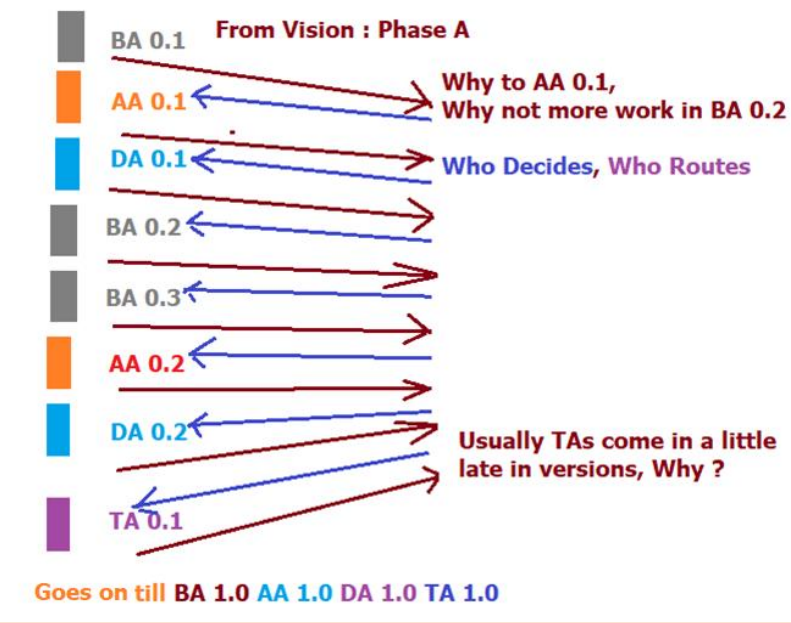
This is what happens, especially though Phases B to G

Requirements and changes thereon, as accepted for current cycle is taken through Phases B to G.



This is where EA decides to move it to
rework (say in BA 0.2) or
move forward (say to AA 0.2)

Nice to Know Box



Quiz Time

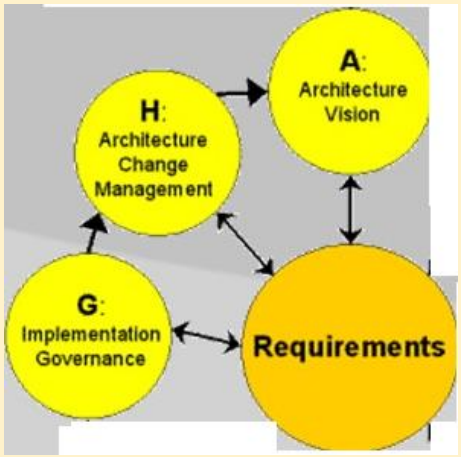
Change in current Phase means :

Work assigned by EA to the corresponding Segment Architect : Yes / No

Yes. Changes in current Phase go through B D A T as required

Change here refers to both fresh Requirement and a slight change in Requirement that is under Architectural work : Yes / No

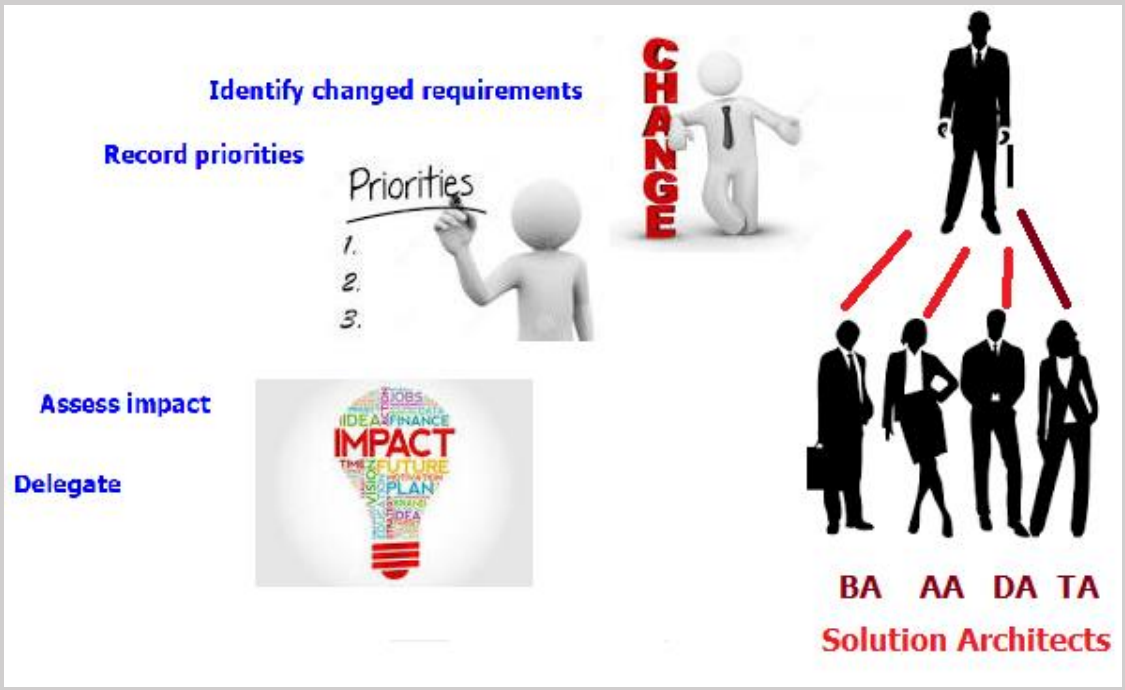
Yes. Slight changes are handled within Phases B to G. Most of the fresh change requests are stored by Requirement Management Phase and then put up for action in Phase H



Nice to Know Box

EA role in Requirement Management

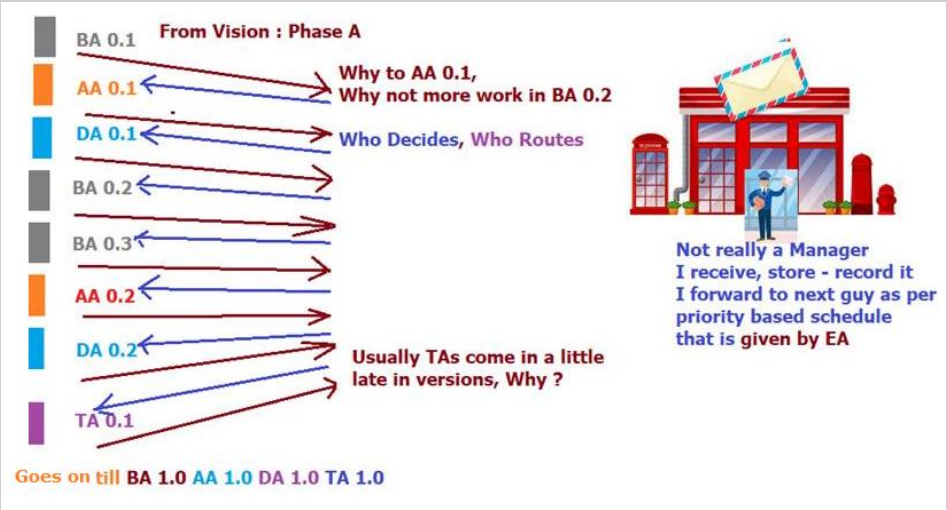
Identify changed requirements
Record priorities
Assess impact of changed requirements
Delegate to implement change in the current Phase



Deliverables in ADM :

Produced by Segment Architect in B D A T
Impact Reviewed with rest of B D A T
Reviewed with relevant Stakeholder
(Impact Analysis and Review may involve Functional and Non-Functional aspects)
Then only approved by EA

EA then decides next course of action



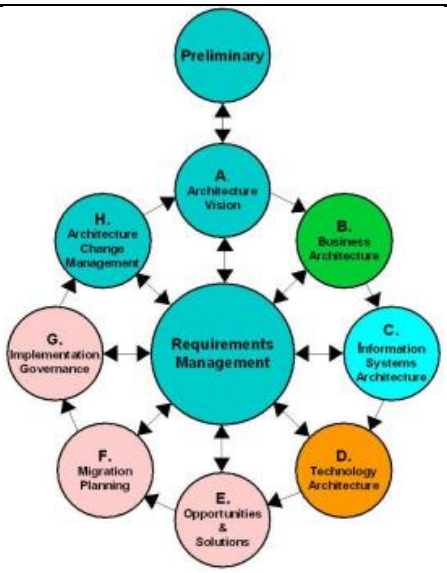
Points to Ponder

Scenario Approach : Where would it fit ?

About the way Requirements are produced in ADM

Requirements are produced, analyzed and reviewed in each of the Phases of the ADM. The Requirements Management Phase describes the process of managing these Architecture Requirements throughout the ADM.

The Requirements Management Phase is central to the ADM – which is why it is shown at the center of the ADM crop circle diagram. This Phase describes a process for Requirements Management, and how that process links to the other Phases of the ADM.



(Only Phase not navigated to Requirement Management Phase is …)

Looking forward to the story of ADM along with the aspect of Requirements

Requirements are not static – they dynamically evolve as we complete each Phase of the ADM, and also between cycles of the ADM. Requirements for Enterprise Architecture, and subsequent changes to those requirements, are identified, stored, and fed into and out of the relevant ADM Phases, and also between cycles of the ADM.

Dealing with changes in requirements is crucial. Architecture deals with uncertainty and change - the "grey area" between what stakeholders expect and what is possible. Architecture requirements are therefore invariably subject to change.

Moreover, Architecture deals with many drivers and constraints which are beyond the control of the Enterprise – such as changing market conditions or new legislation - which can produce changes in requirements in an unforeseen manner.

TOGAF emphasizes that the Requirements Management process itself does not dispose of, address, or prioritize requirements, as this is done within the relevant Phase of the ADM.

The Requirements Management Phase is merely the process for managing requirements throughout the overall ADM.

Nice to Know Box

Although functional and quality requirements specify the goal, constraints limit the (Architectural) solution space. Stakeholders should therefore not only specify requirements, but also constraints. **The version slices (such as moving through 0.2 to 1.0 in each of BA. DA. AA and TA) gives opportunity to engage with Stakeholders at different levels and different angles to get the best out of them**.

Possible **constraint categories** could be the following :

– Technical constraints, such as platform, reuse of existing systems and components, use of standards; - Which Viewpoint of Stakeholders and from which Stakeholders ?

– Financial constraints, such as budgets; Which Viewpoint of Stakeholders and from which Stakeholders ?

– Organizational constraints, such as processes, skill (or lack of skill) of employees,
formal rules and policies; Which Viewpoint of Stakeholders and from which Stakeholders ?

– Time constraints, such as deadlines Which Viewpoint of Stakeholders and from which Stakeholders ?