Enterprise Layers of Architecture

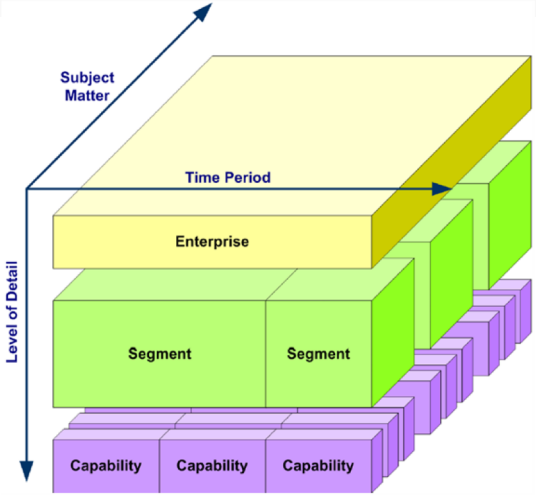
Kim Horn

Version 1.0 **1 June 2015**

Agenda

- Why Layers of Architecture?
- The 3 layers:
  - Enterprise Process
  - Segment / Portfolio Process
  - Solution Process
- Technical Architecture Integration
- Waterfall to Iterative to Stream of Work
- Artefacts
- 2 Minds

Architecture – 3 Main Types



Enterprise Architect

- Responsible for cross-enterprise standards, roadmaps, principles and frameworks
- Alignment between enterprise and business unit roadmaps
- Maintains and develops overall architectural competency
- Defines common solution architecture process and supports integration with project processes
- Ensures effective architecture governance
- Highest technical authority within the enterprise
- Typically 3-5 year horizon

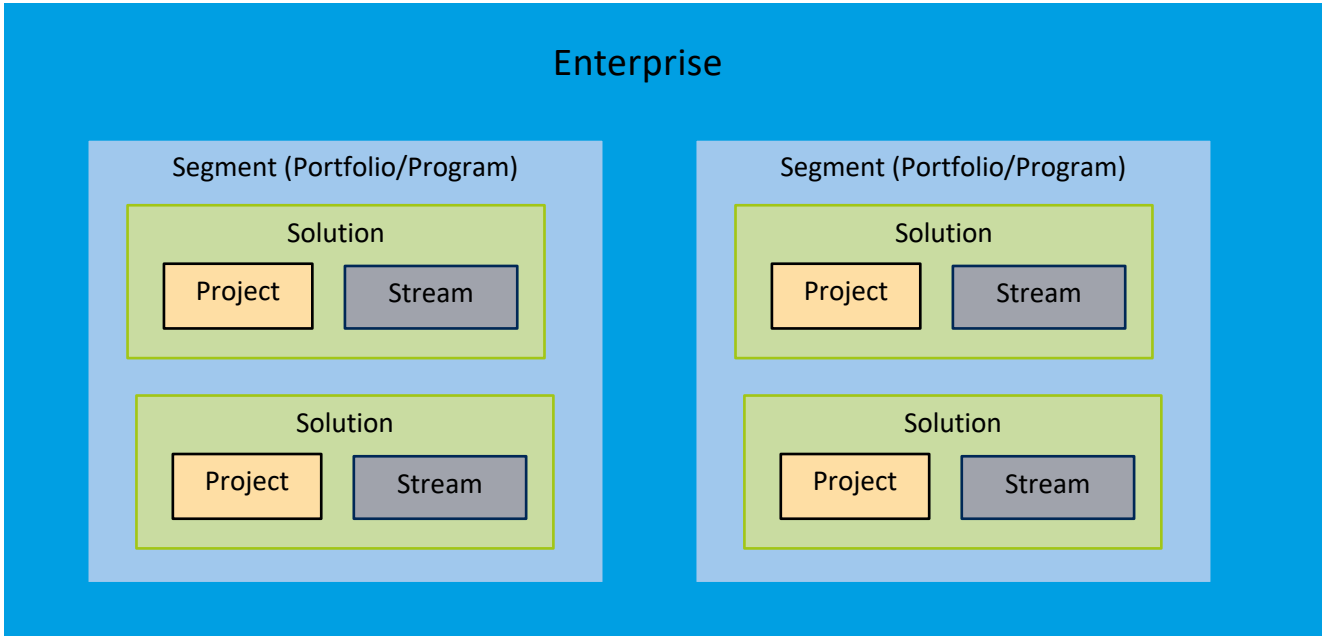
Segment - Portfolio, Program, Domain Architect/LOB Manager

- Strong strategic capability and business domain knowledge.
- Expert in a given business or technical segment (e.g. applications, infrastructure, security, formation, networks, etc.)
- Set and enforce segment standards and principles e.g. security
- Develop, manage and communicate the segment intellectual property (IP) i.e. roadmaps, standards, reference architectures, etc.
- Supports Solution Architectures via provision of management, IP, direction and reviews
- Highest technical authority within a segment
- Typically 1-3 year horizon

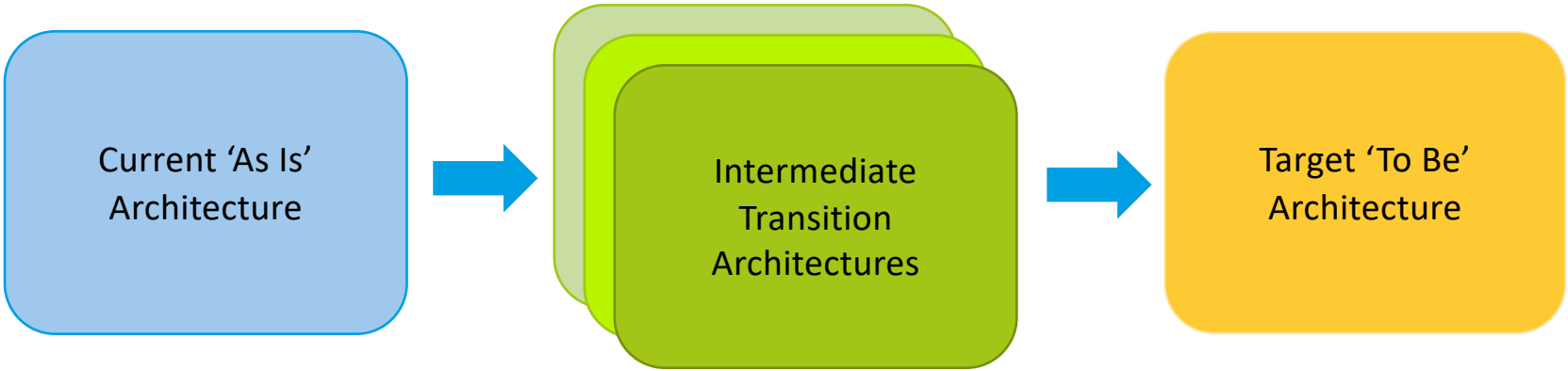
Capability - Solution Architect

- Responsible for ensuring the project meet key “quality” requirements;
- Responsible for overall solution architecture coordinating the activities of other architects and specialists;
- Ensures solution is developed in accordance with roadmaps & standards;
- Highest technical authority within a project;
- Operates within project timeframe.
- Continuous Stream or project focus

3 Layers - Enterprise to Segment to Solution (Project or Stream)



Architecture States



Architecture - Concurrent & Continuous Ways of Working

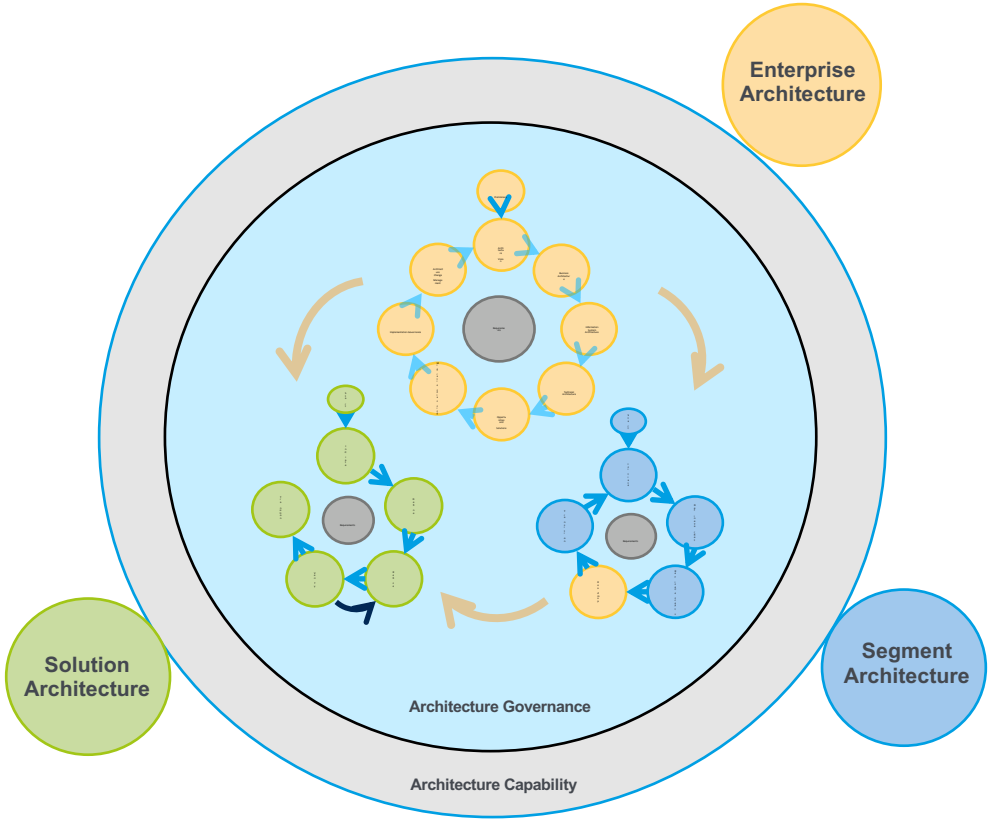
3 Core 'Ways of Working':

- Enterprise
- Segment / Portfolio
- Solution

2 Cross Cutting Disciplines

- Governance
- Capability

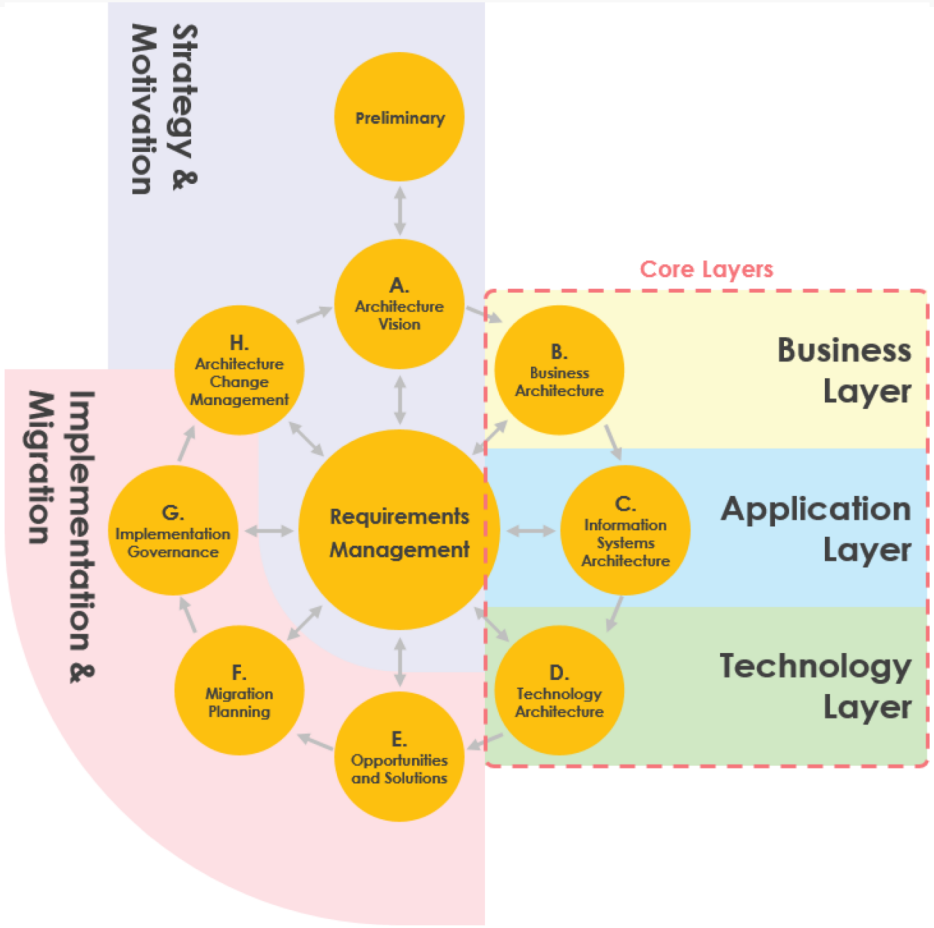
- All run concurrently and continuously
- Requirements is the core of all 3



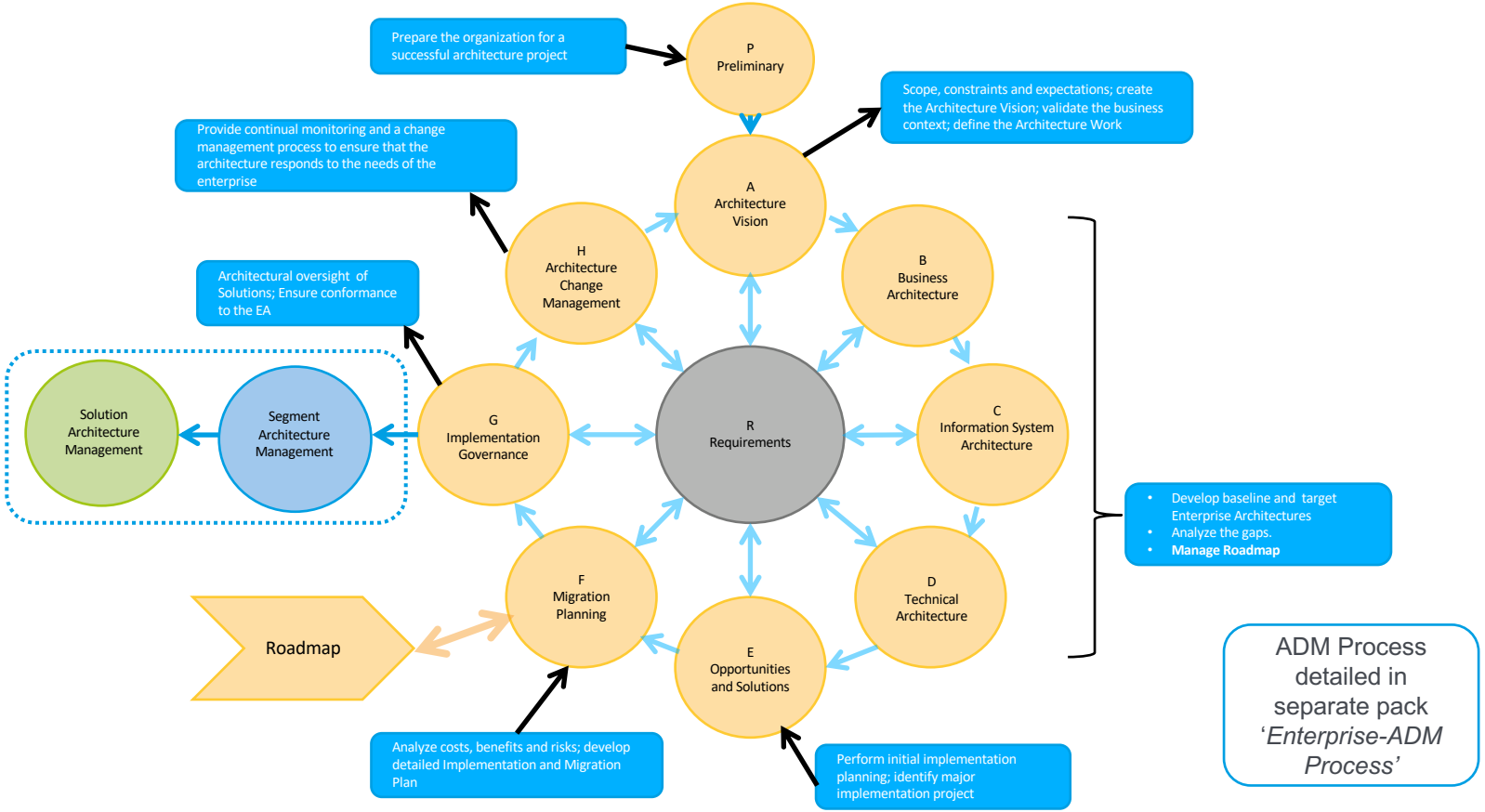
Enterprise Process
TOGAF

Enterprise Architecture

- Enterprise Architecture is Strategy
- It aligns the Architectures to the Business Strategy
- Understand what we have and how we use it
- Understand how we execute the mission
- Align programs, projects, businesses, companies.



Enterprise Architecture – TOGAF ADM



Segment Process

Segment Architecture

Segment Architecture is about:

- Architecting across a set of solutions, within a particular domain or business segment, to deliver capability increments consistent with the Enterprise and Business direction;
- Providing deep Subject Matter Expertise in the particular business or domain area;
- Provide subject relevant architectural collateral, at a more detailed level than EA;
- Engaging with the business or domain closely to maintain the relationship;
- Help guide and align the Segment Architecture Roadmap with the Project Roadmap.
- Help guide and align the Enterprise Architecture Roadmap with the Segment Roadmap.
- Manage a shorter term focus than EA;
- Provide an Intentional Architecture to allow Solutions to proceed;

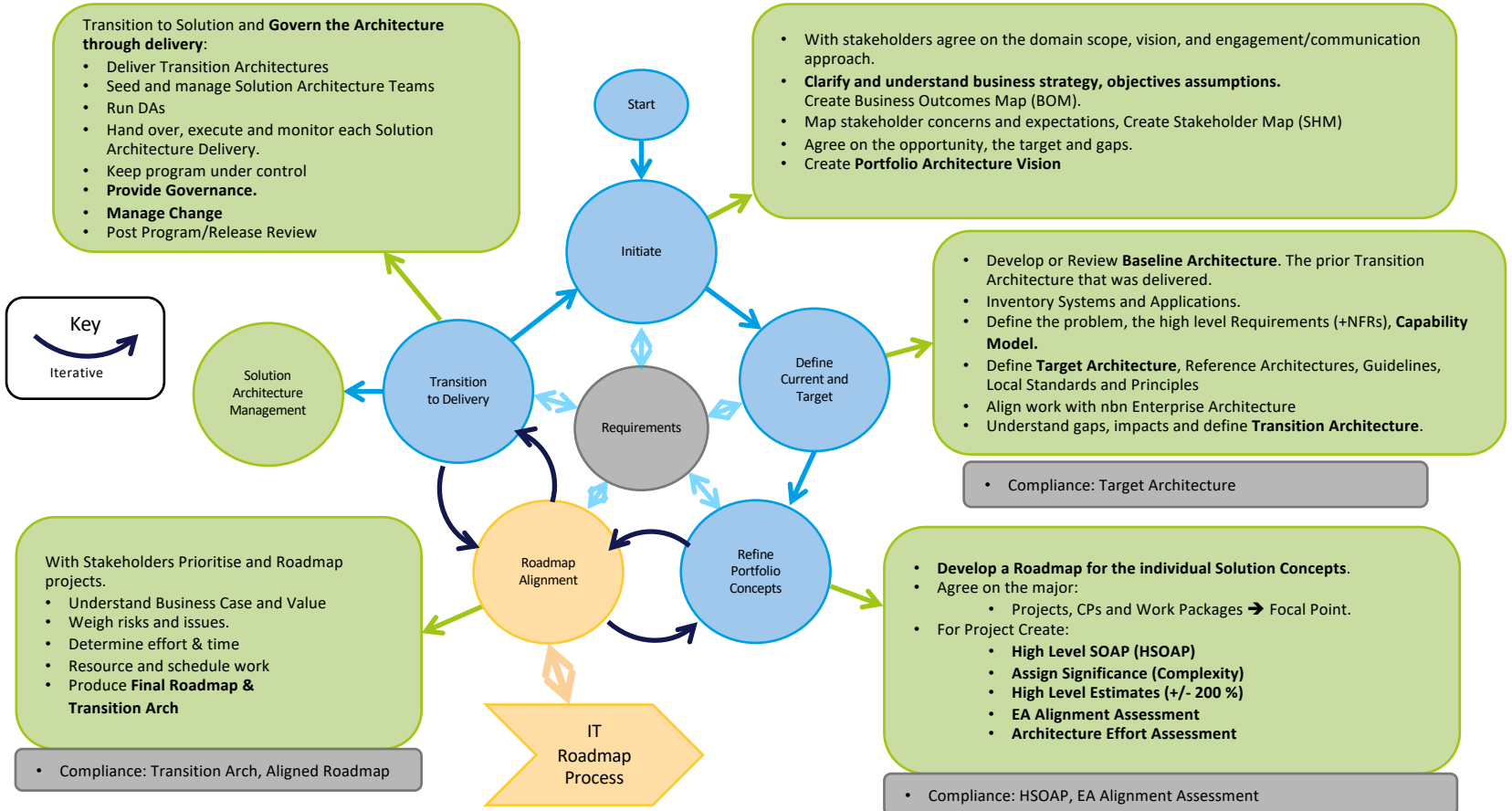
Segments provide the way to manage IT investments and this process links with an IT Road mapping process.

- Starts off the conceptual work that will result in an Enterprise Architecture aligned Target Architecture.
- Road mapping may result in a number of Transition Architecture States, implemented as Initiatives/Programs/Releases a portfolio of projects.

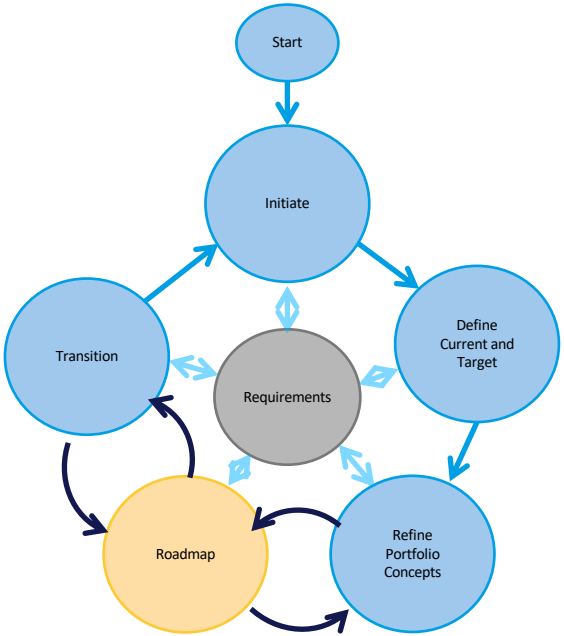
Two main segment types of work:

1. **Lines of Business**: the projects could be organised by Program, Initiatives, LOB Segment, or Release. Portfolio Architecture work is transitioned to Solution Architecture for delivery of capability.
2. **Technical Domains:** Capabilities include: Infrastructure, Security, Integration, Data and Platforms.

Segment Architecture Management Process – Generic



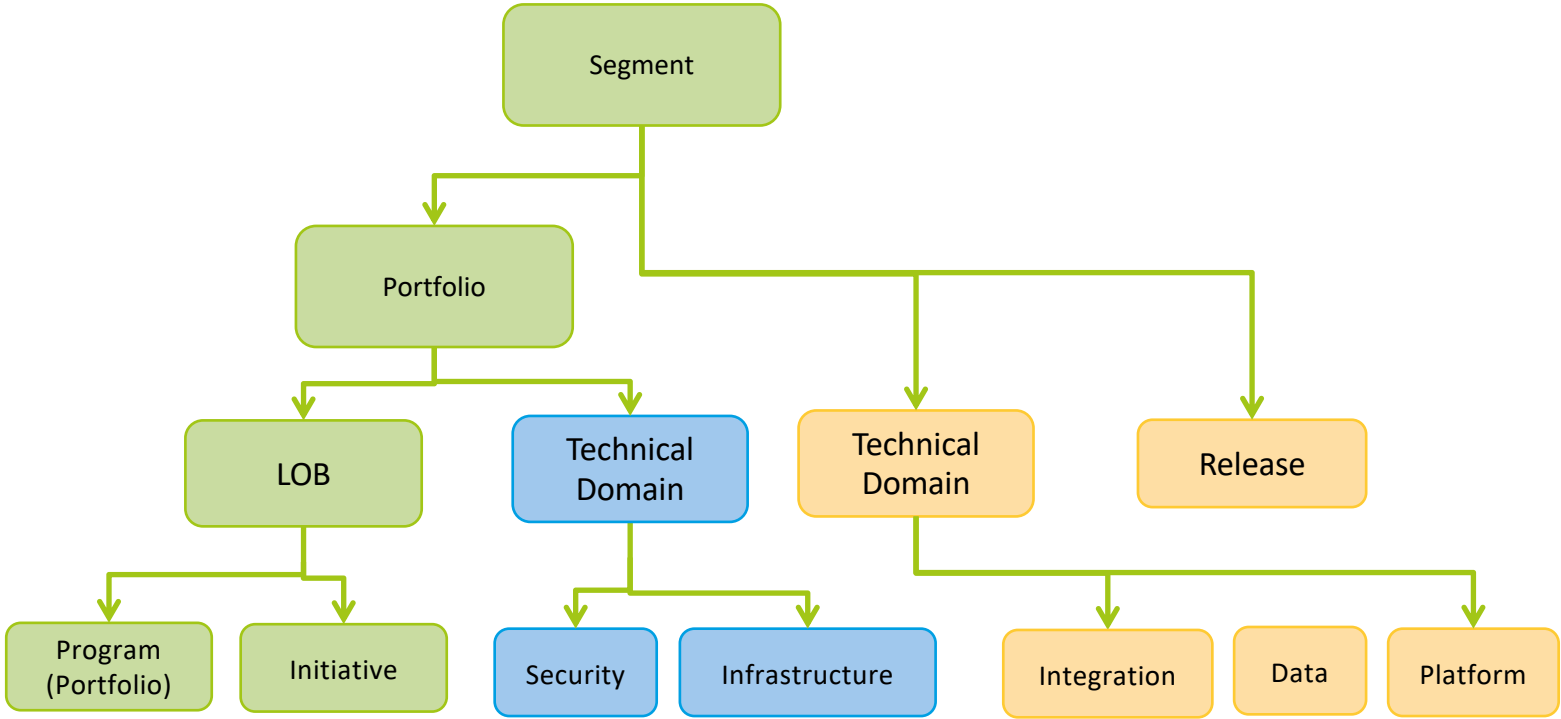
Process Activities



The circles represent major groupings of activities only:

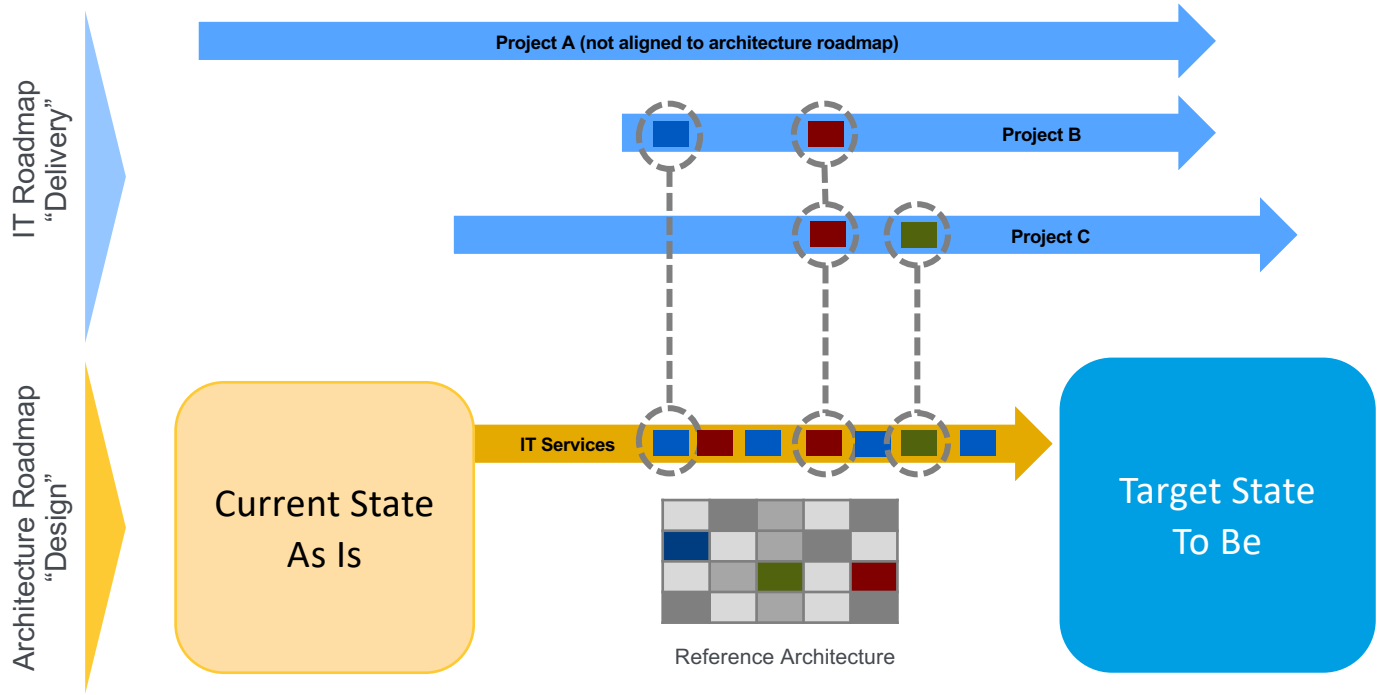
- They are not definitive phases in a process;
- They are not gates;
- They suggest things to be done, some early some later;
- They all can be iterative, some (blue lines) can be iterative.
- Iterative delivery of small increments is recommended.
- Not all steps are done by all domains.
- Not all activities in each step are done
- The term “Refine” is used to reinforce the Iterative nature of this work.

Segment Architecture – Many Faces & Recursive



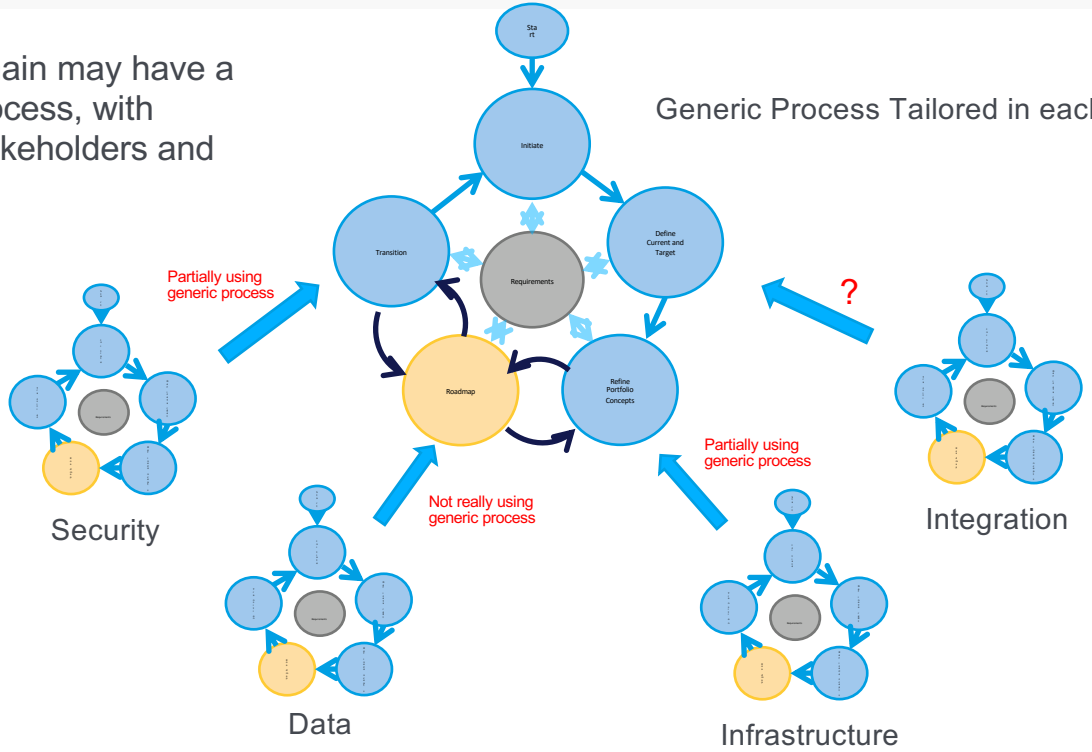
IT Portfolio Roadmap Vs Architecture Roadmap

IT Roadmap, uses projects, or stream releases, to drive progress to the agreed target state defined in the Architecture Roadmap



Generic Process may not apply well across all domains

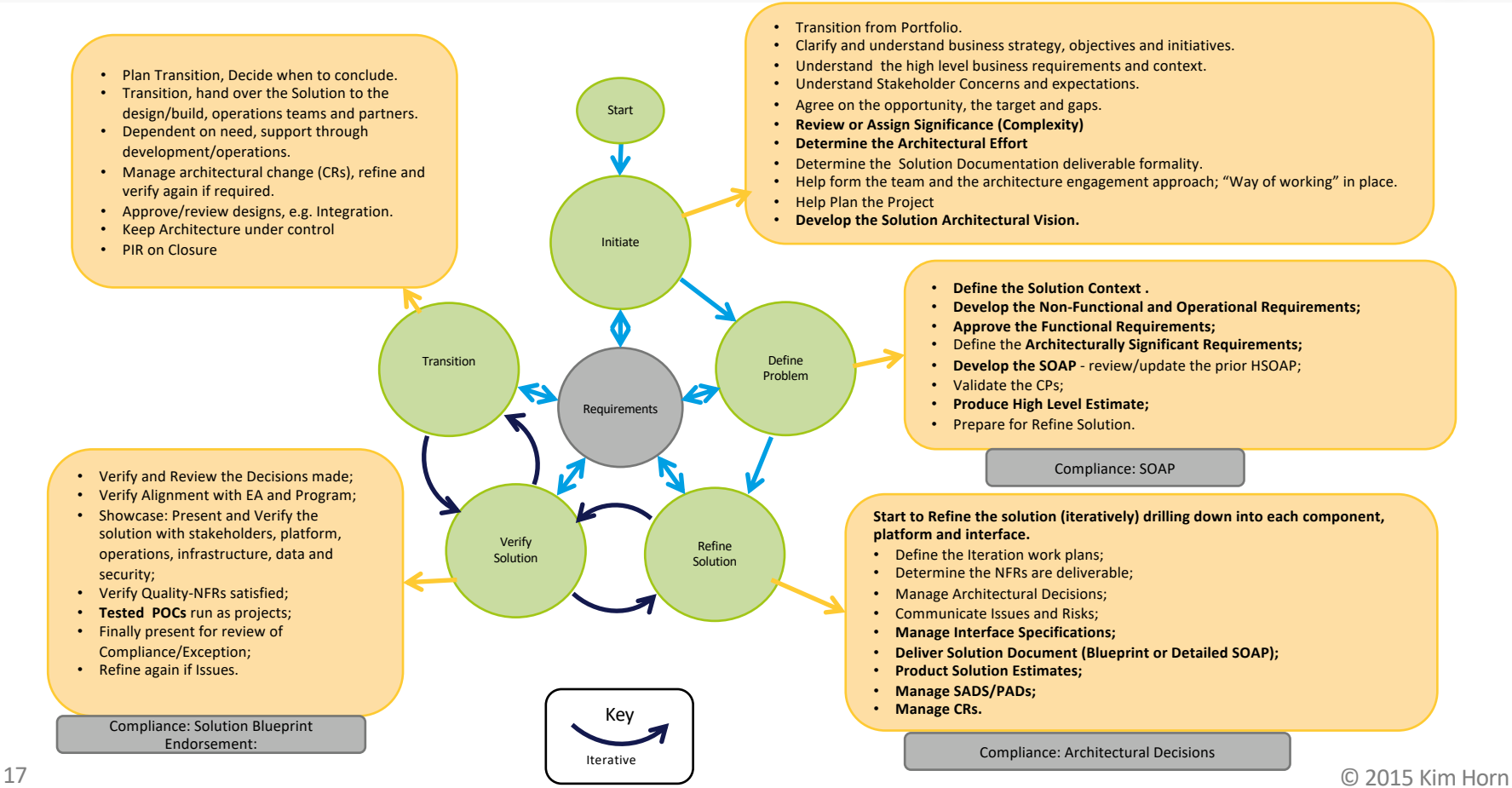
Each Domain may have a unique process, with unique stakeholders and handoffs.



Each Domain may have a unique process, with unique stakeholders and handoffs.
Each Domain may also have other Disciplines independent of the Generic Process.

Solution Process

Solution Architecture Management Process – 6 Activities



- Transition from Portfolio.
- Clarify and understand business strategy, objectives and initiatives.
- Understand the high level business requirements and context.
- Understand Stakeholder Concerns and expectations.
- Agree on the opportunity, the target and gaps.
- **Review or Assign Significance (Complexity)**
- **Determine the Architectural Effort**
- Determine the Solution Documentation deliverable formality.
- Help form the team and the architecture engagement approach; "Way of working" in place.
- Help Plan the Project
- **Develop the Solution Architectural Vision.**

- Plan Transition, Decide when to conclude.
- Transition, hand over the Solution to the design/build, operations teams and partners.
- Dependent on need, support through development/operations.
- Manage architectural change (CRs), refine and verify again if required.
- Approve/review designs, e.g. Integration.
- Keep Architecture under control
- PIR on Closure

- **Define the Solution Context .**
- **Develop the Non-Functional and Operational Requirements;**
- **Approve the Functional Requirements;**
- Define the **Architecturally Significant Requirements;**
- **Develop the SOAP** - review/update the prior HSOAP;
- Validate the CPs;
- **Produce High Level Estimate;**
- Prepare for Refine Solution.

Compliance: SOAP

- Verify and Review the Decisions made;
- Verify Alignment with EA and Program;
- Showcase: Present and Verify the solution with stakeholders, platform, operations, infrastructure, data and security;
- Verify Quality-NFRs satisfied;
- **Tested POCs** run as projects;
- Finally present for review of Compliance/Exception;
- Refine again if Issues.

Compliance: Solution Blueprint Endorsement:

Start to Refine the solution (iteratively) drilling down into each component, platform and interface.

- Define the Iteration work plans;
- Determine the NFRs are deliverable;
- Manage Architectural Decisions;
- Communicate Issues and Risks;
- **Manage Interface Specifications;**
- **Deliver Solution Document (Blueprint or Detailed SOAP);**
- **Product Solution Estimates;**
- **Manage SADS/PADs;**
- **Manage CRs.**

Compliance: Architectural Decisions

Continuous Architecture Practice Principles

Perform architecture work "just in time". identify and discuss architectural issues with your team, and then prioritize architectural work with any other work.

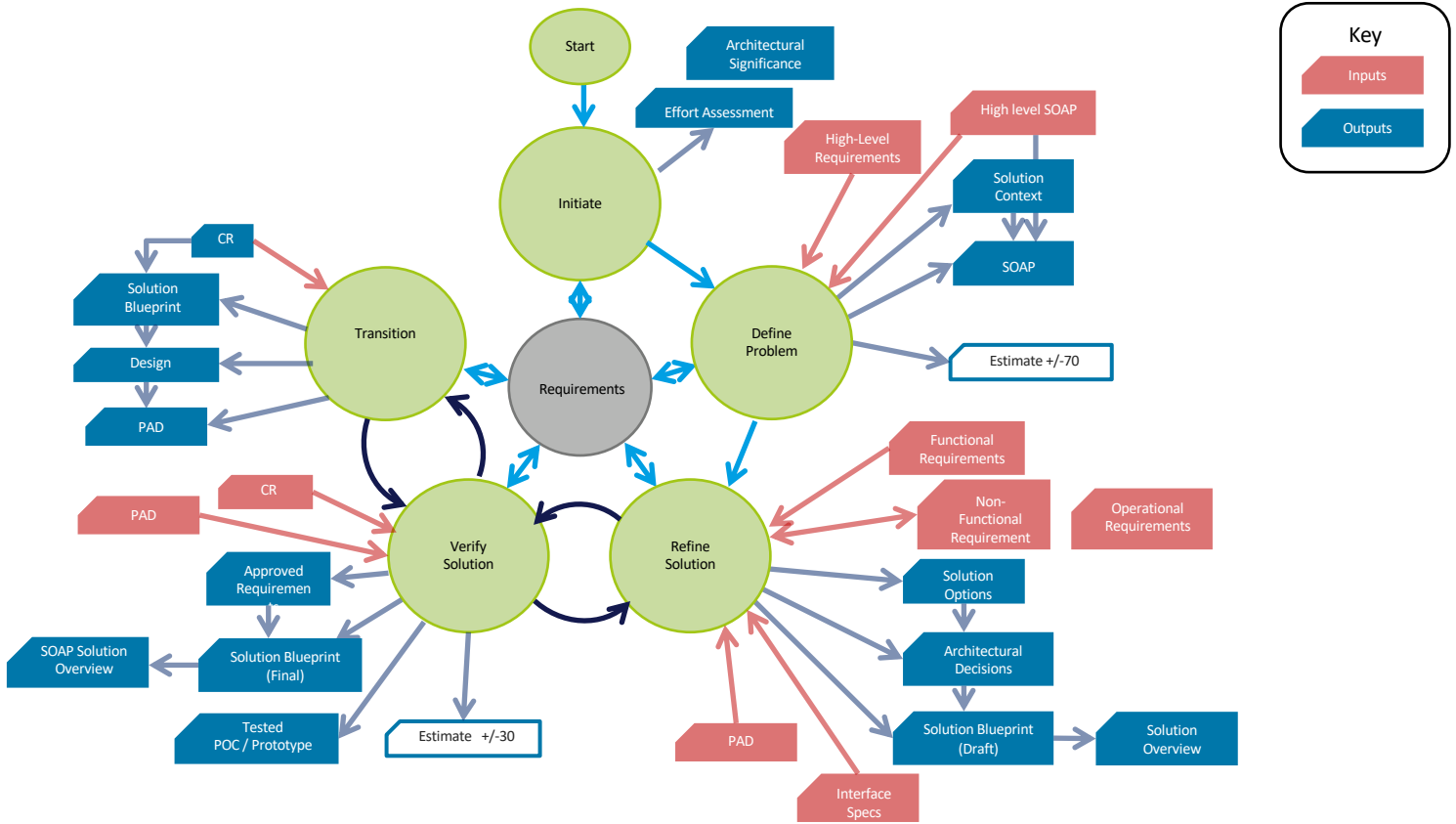
- Refine the architecture iteratively and incrementally, from the base **Definition**;
- Provide enough Runway for development to proceed;
- Defer architectural issues to handle them "just in time" enables the architecture to *evolve over time*.
- Base your initial priorities on mitigating technical risk rather than creating value. Provides an initial Architecture **Definition** that is then **Refined**;
- Harvest ***Emergent Architecture*** feed back to ***Intentional Architecture***;

Document key architectural decisions and outstanding issues.

Implement and test key capabilities as a way to address architectural issues. Resolving architectural issues typically requires not only architectural brainstorming, but also associated prototyping.

- Prototypes are small throw away builds that validate the assumptions behind the architecture or mitigate risks.

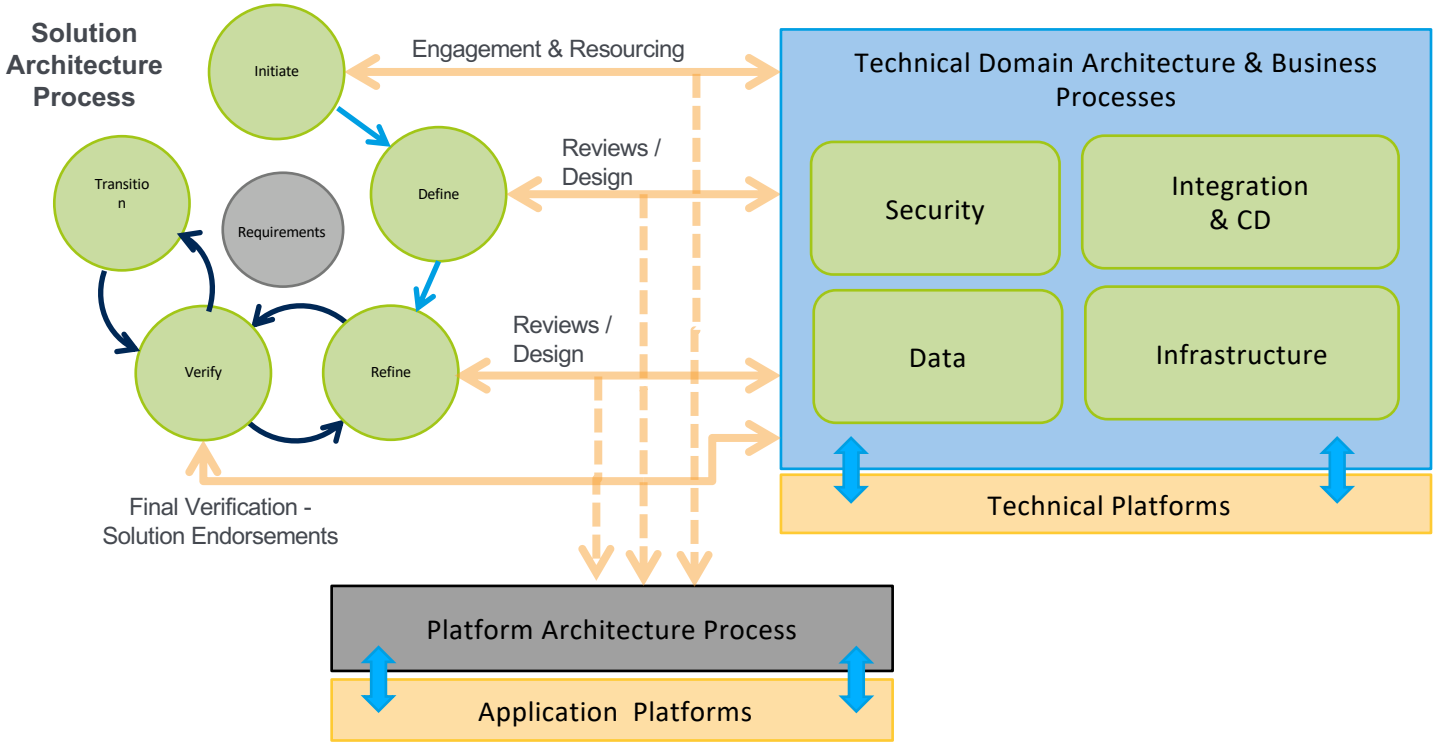
Solution Architecture Management Process Artefacts



Technical Integration

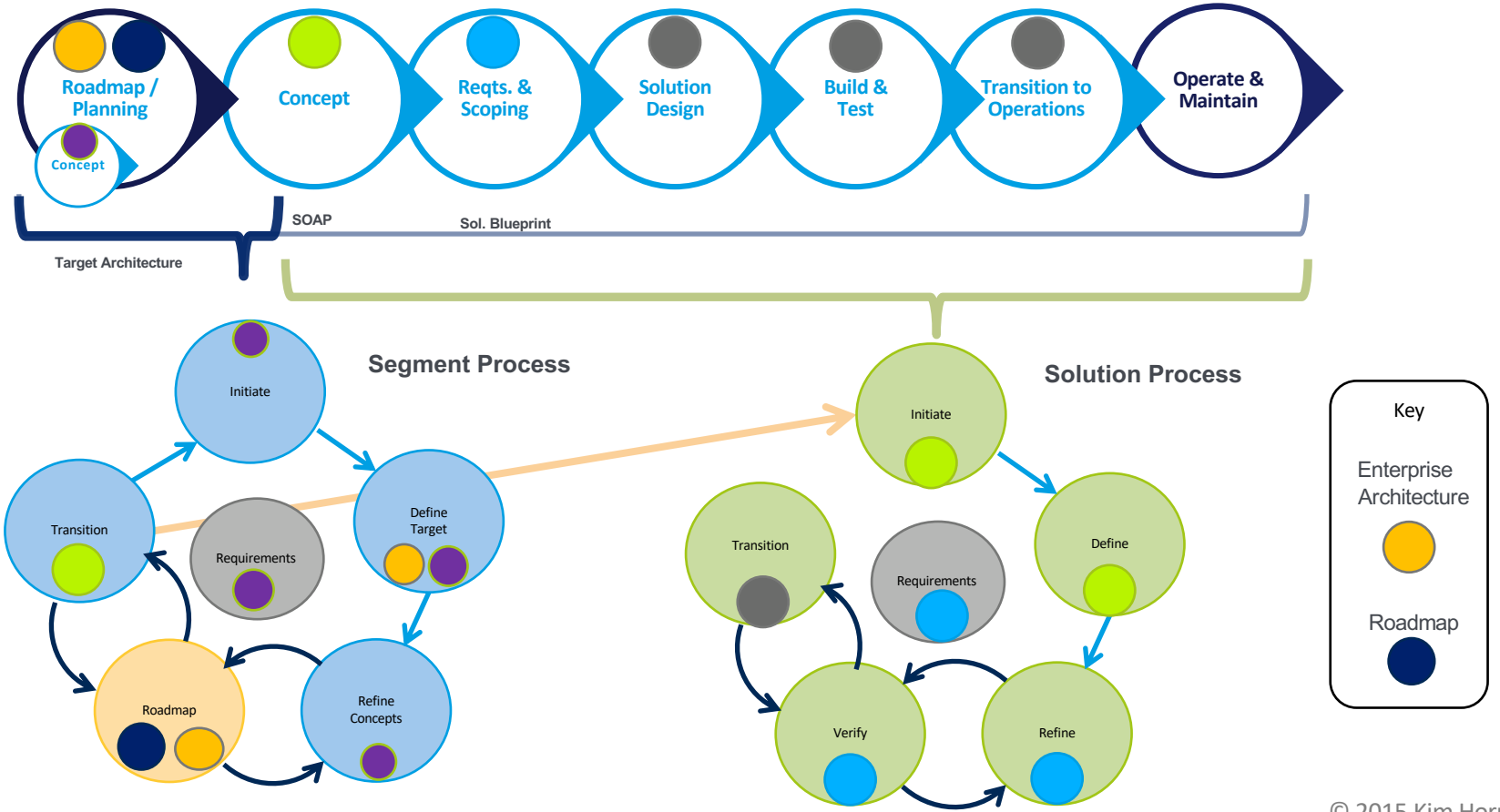
Solution, Technical Platform Architecture Integration

- Solution Architects get support from the Technical Domains; request reviews, help with design and final endorsements;
- Solution Architects also get support from Platforms;
- Each Technical Domain has their own processes to support the solutions;
- Some domains have Business counterparts, e.g. Security and Data.



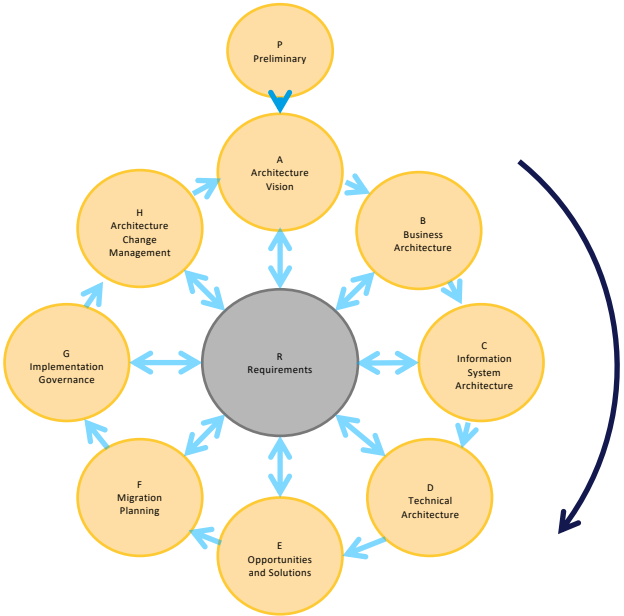
Waterfall To Iterative To Stream of Work

Architecture, Roadmap and Waterfall Intersection

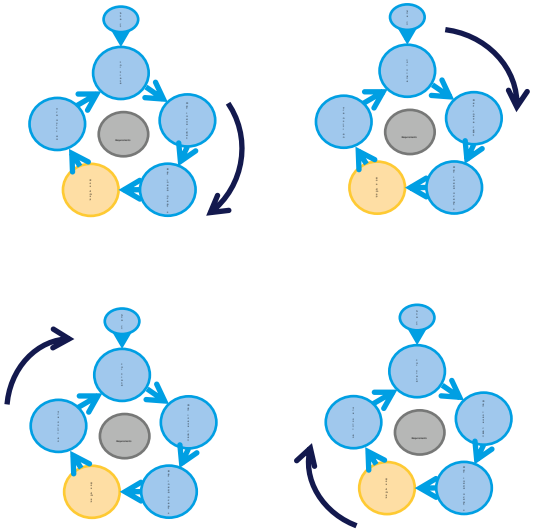


Iterative Architecture From Enterprise to Portfolio

Enterprise Architecture Development



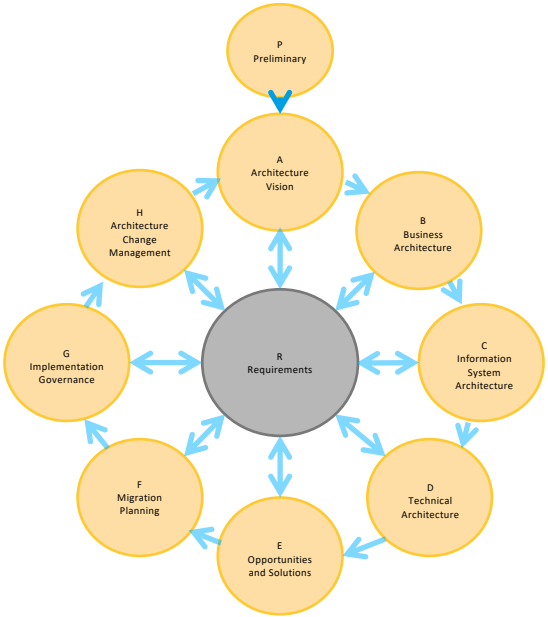
Segment Architecture Development



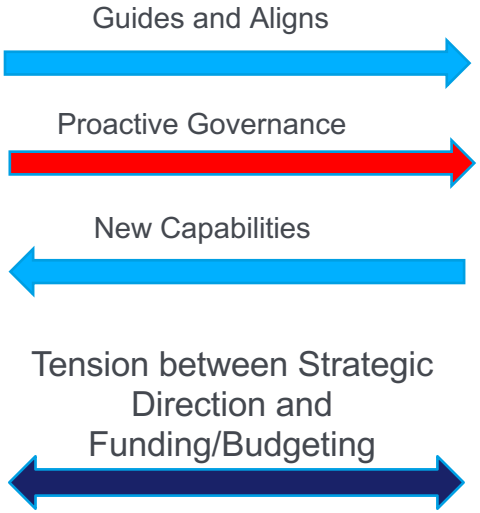
- EA may iterate by focusing on particular scope, detail, or segments (domains).
- One EA Iteration may focus on one Segment Portfolio, or a number of Portfolios concurrently.
- Portfolios will most likely run iterations asynchronously but ideally all synchronise into one EA Iteration.

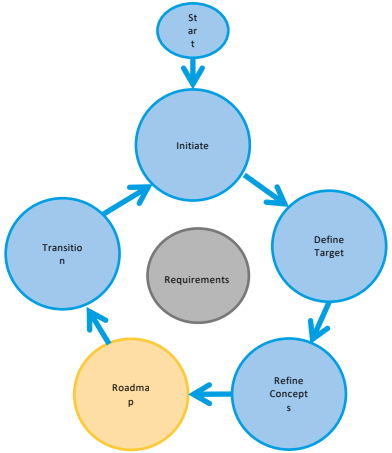
Enterprise to Portfolio Tension

Enterprise Architecture Development



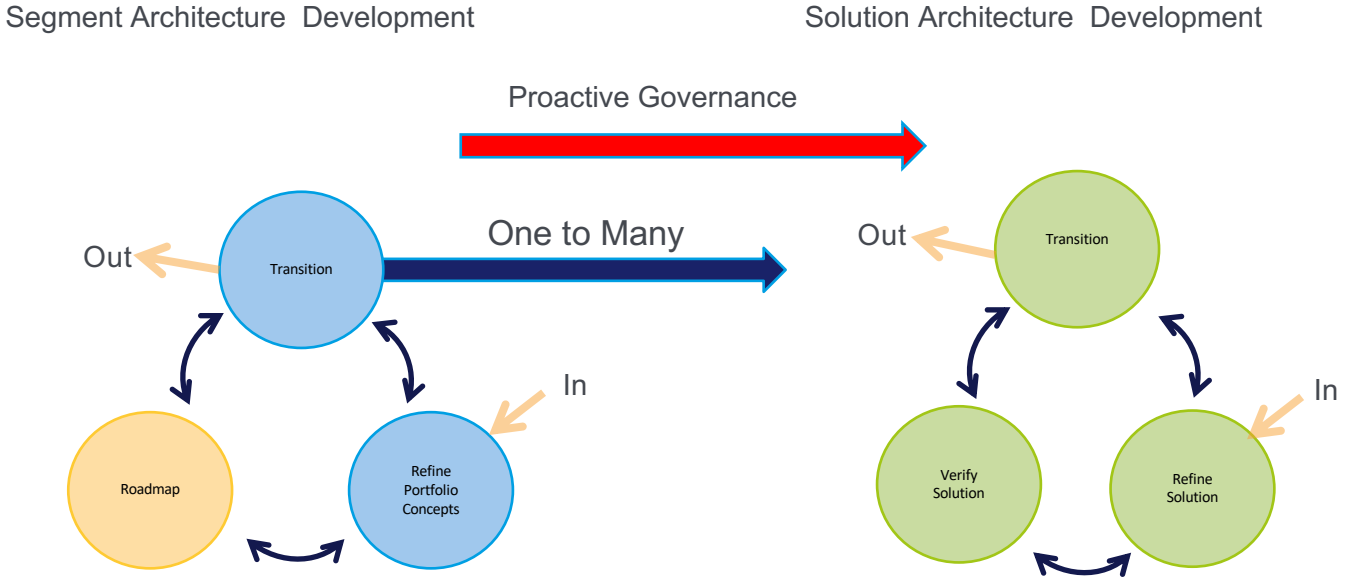
Portfolio Architecture Development





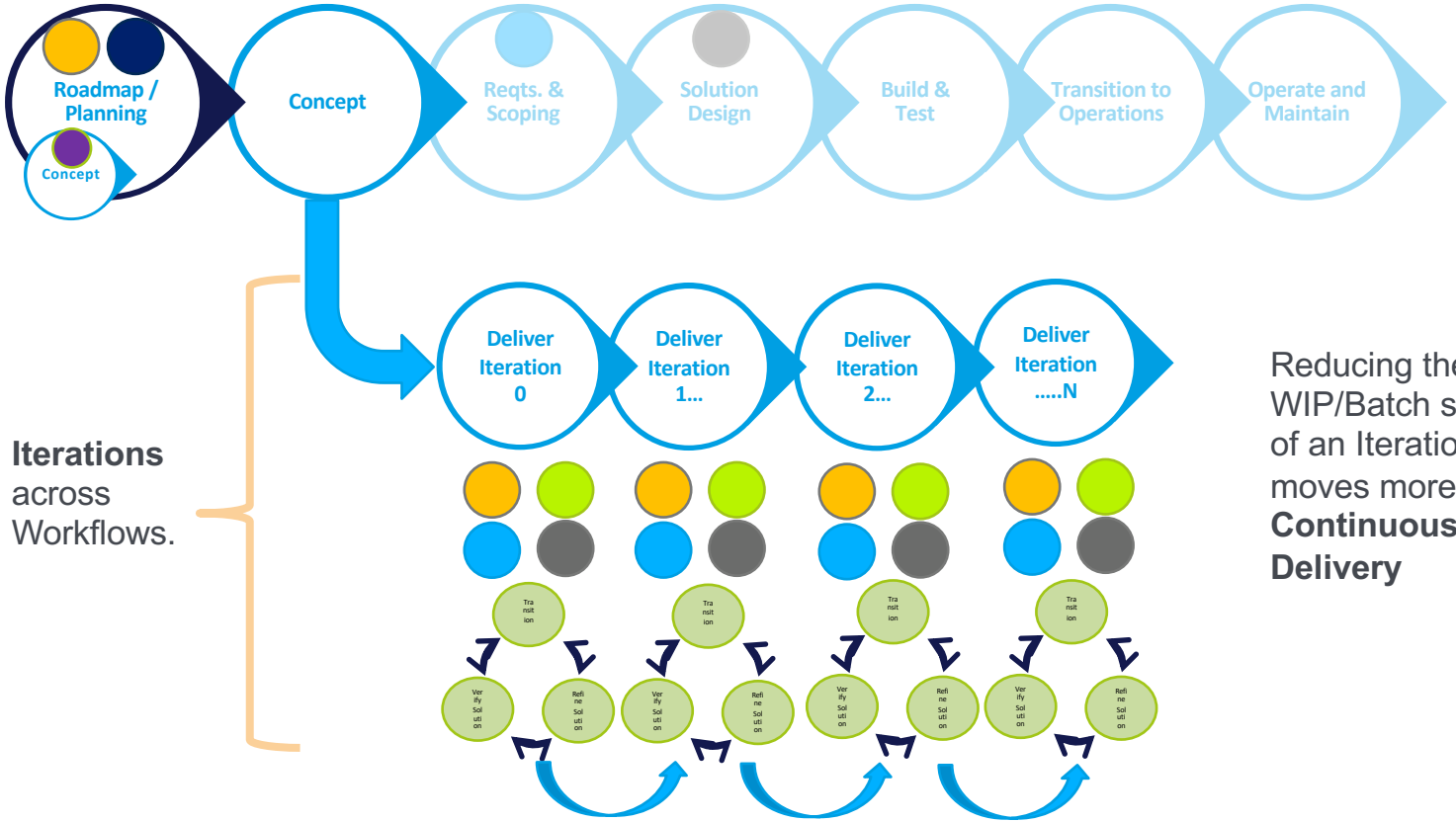
Governance: Portfolio Architecture proactively ensures the currency and relevancy of an organisation's Enterprise Architecture and therefore the design integrity of the organisation itself.

Iterative Architecture From Segment to Solution



- Software Architecture processes are best conducted iteratively. At each iteration the prior results are reviewed and refined, in an evolutionary manner. This approach discovers flaws in the architecture early and allows change with minimal impact.
- Furthermore, requirements change through emergent understanding is catered for with reduced impact.
- Portfolio Architecture may Transition Multiple Solutions at one time, each at different stages.

Supporting Iterative and Waterfall IT Processes

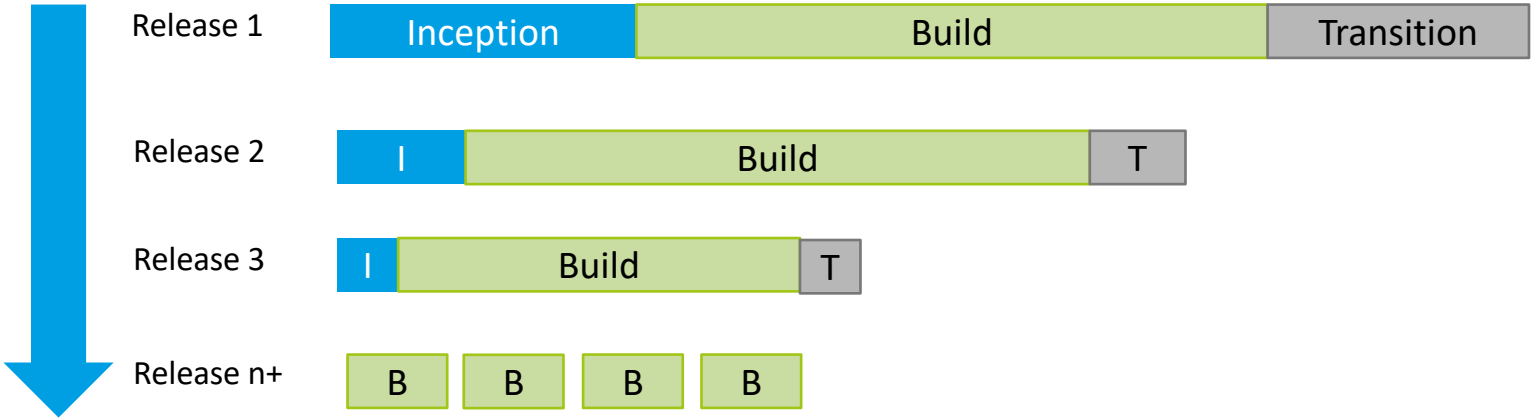


Iterations across Workflows.

Reducing the WIP/Batch size of an Iteration moves more to **Continuous Delivery**

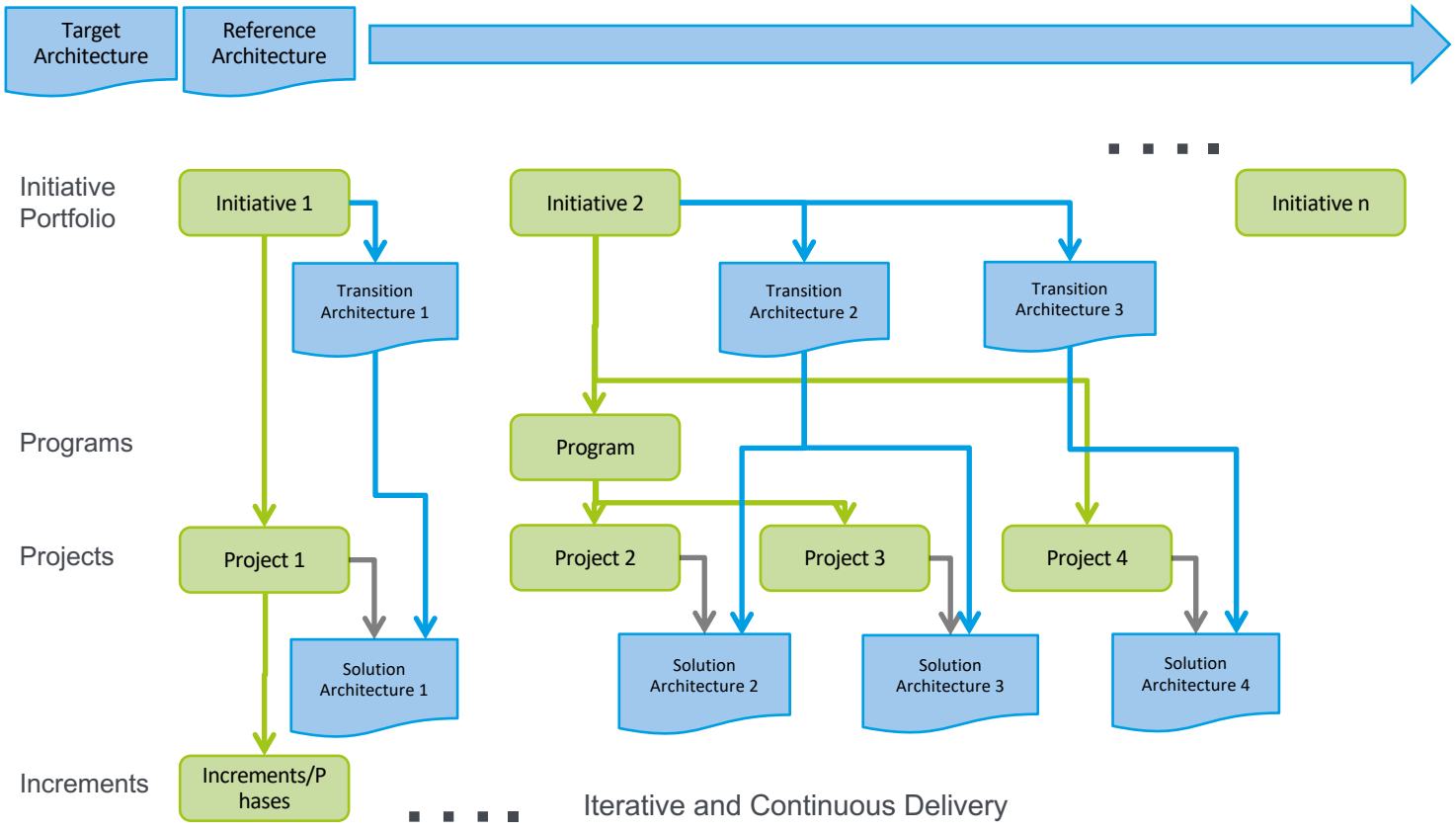
Evolving Project Types

- New Projects require a more traditional model, with some governance, controlled phases, and more upfront activity;
  - Some formality to get things going;
- With successive releases the phases start to disappear;
- Until finally work is shorter continuous iterations.

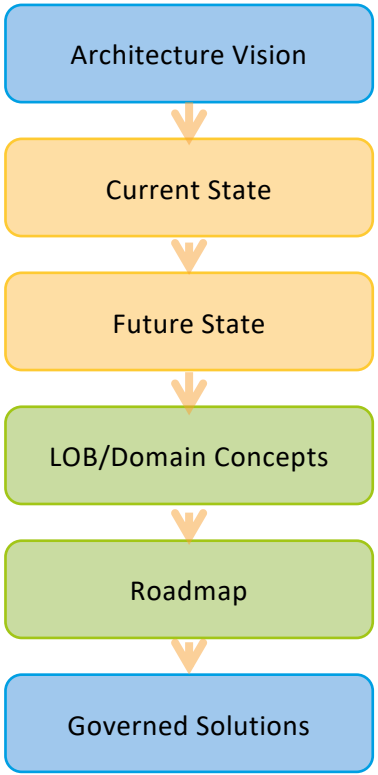


Artefacts

Initiative Portfolio Delivery via Programs/Projects with Architectures



Work Products over Lifecycle



Create a Architecture vision of the work to be done and program structure

Document or Review the current state ‘as is’ architecture across the Enterprise and in each domain or LOB

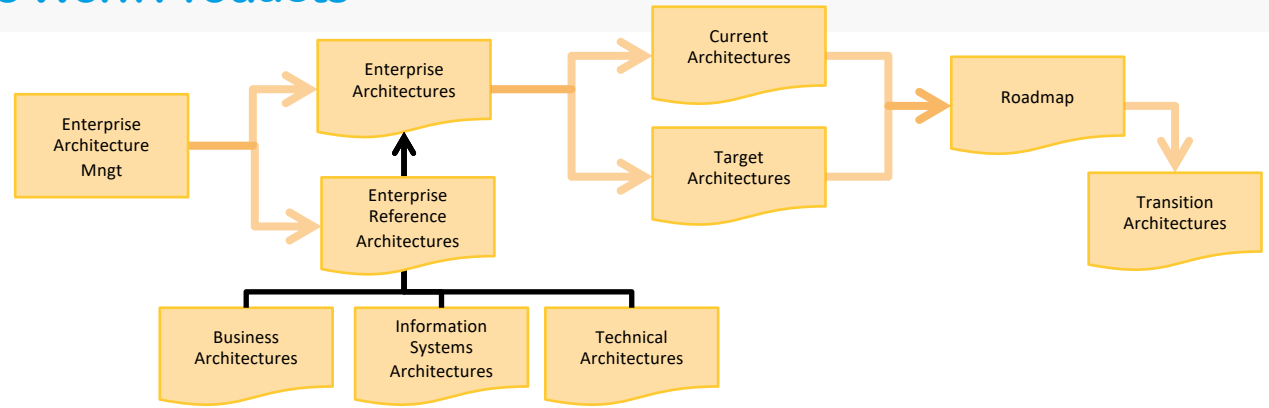
Develop a future state ‘to be’ target, transition and reference architectures across the Enterprise and in each domain or LOB

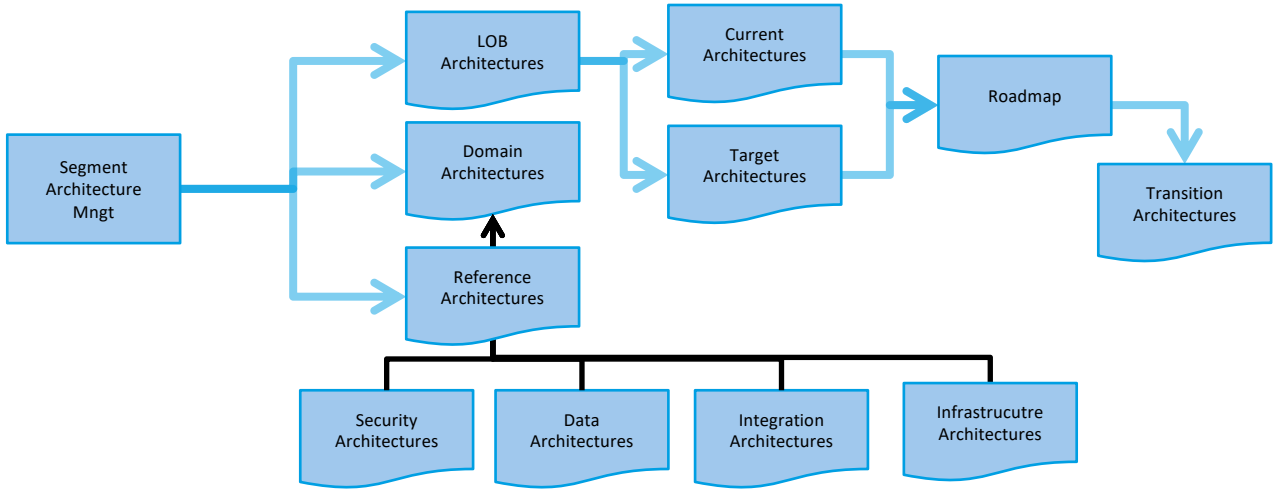
Create a collection of work concepts (projects) and transition architectures

Create a Roadmap to get from Current State to Future State.
Prioritise tension between Strategic Direction and Funding/Budgeting.

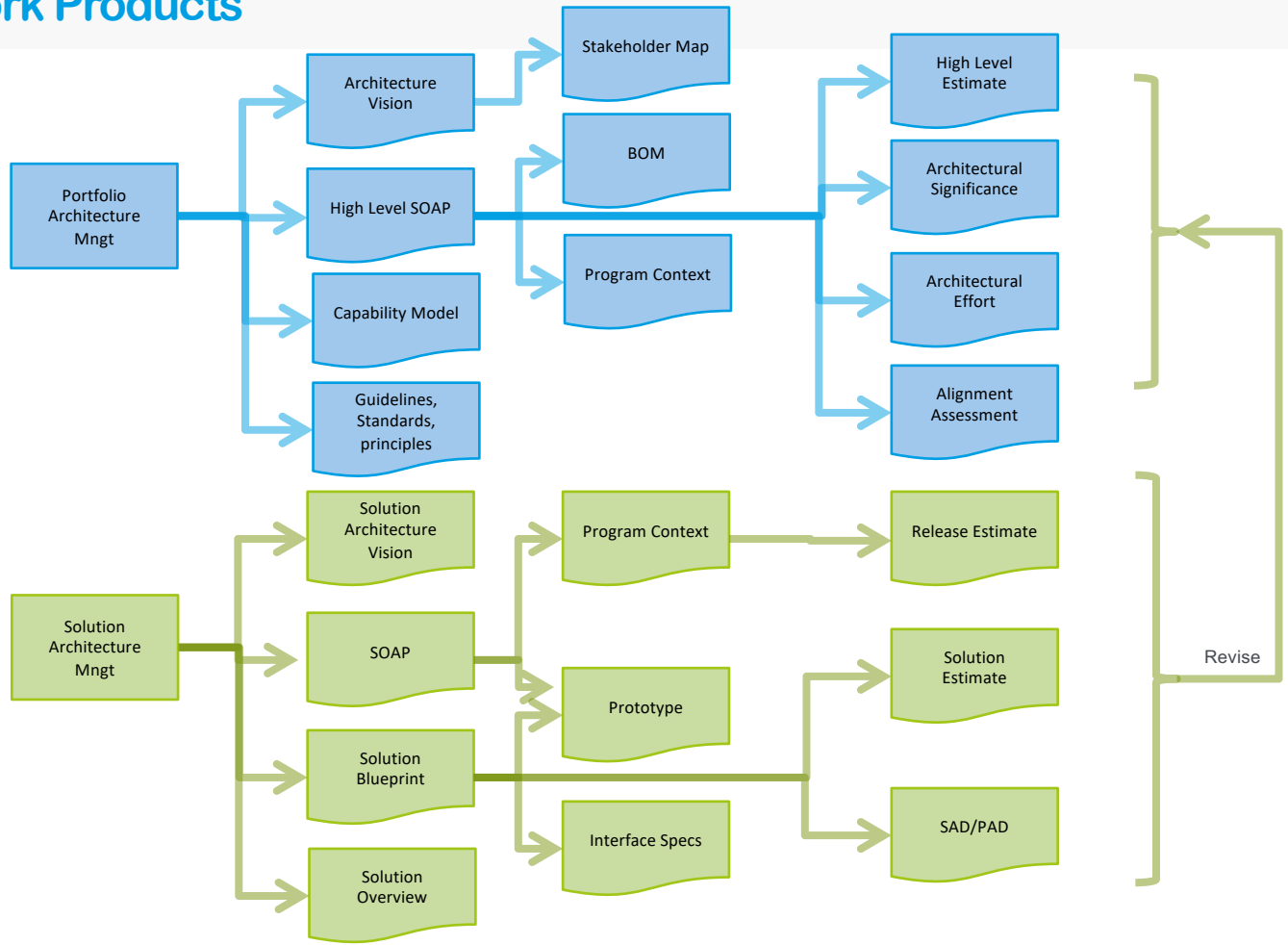
Hand over, govern and manage the individual solution architectures.

Architecture Work Products

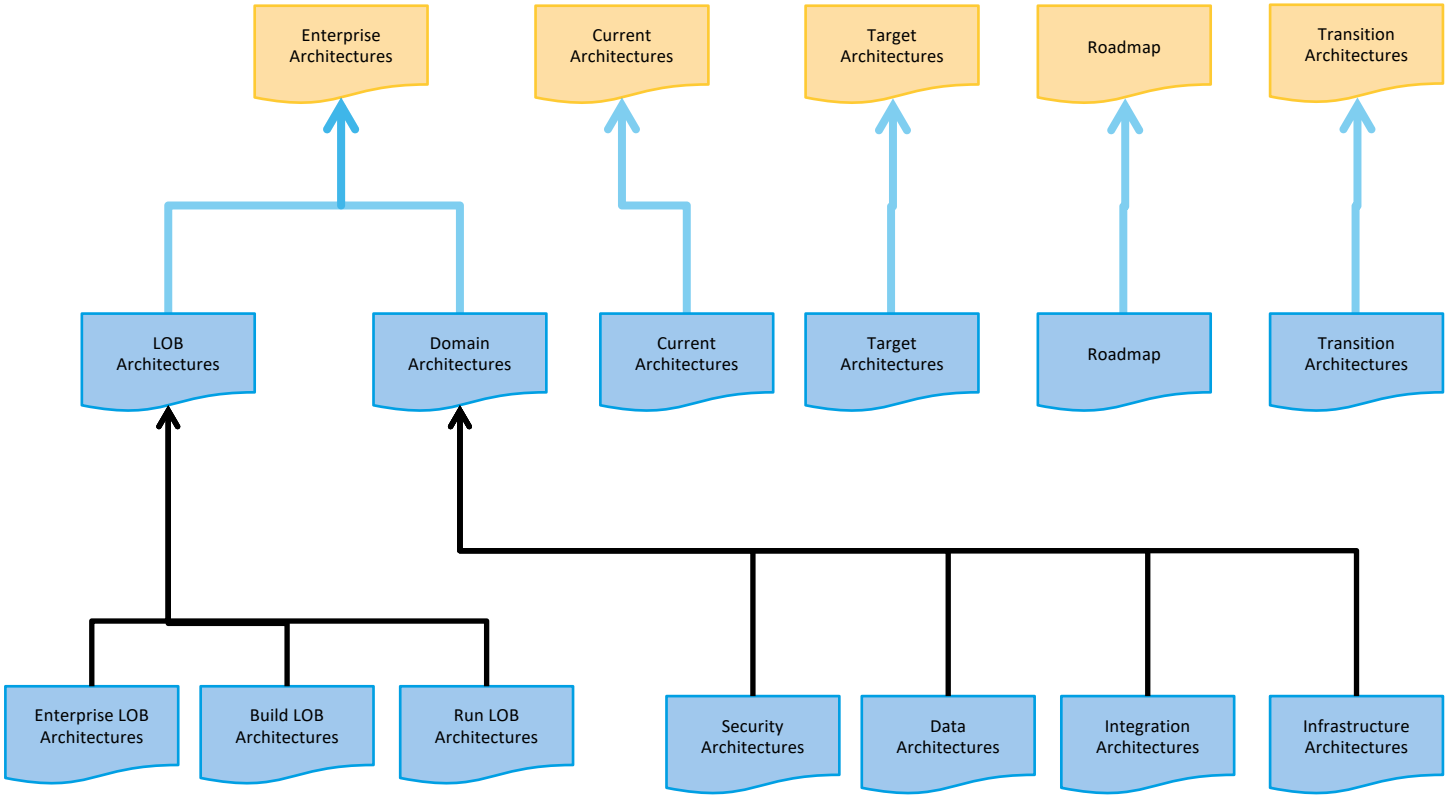




Other Work Products



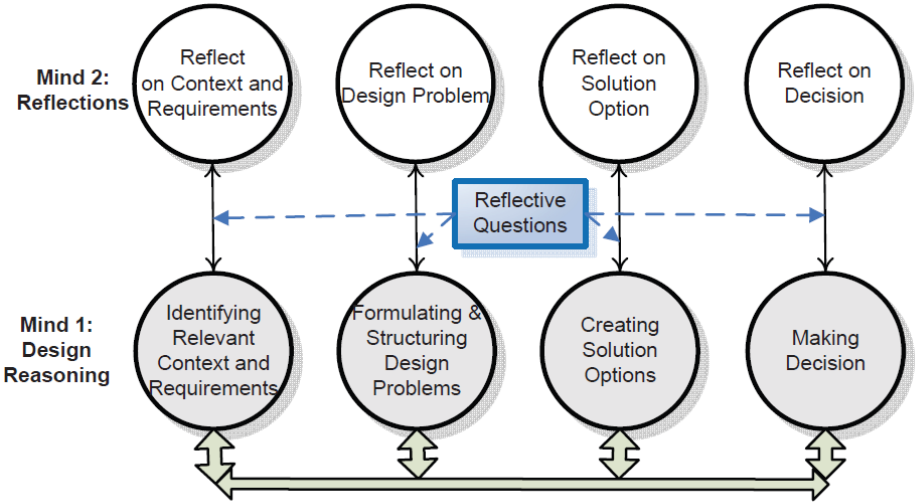
LOB To Enterprise Architecture Work Products



2 Minds

2 Minds: Psychological / Decision Modelling Background

The processes proposed are based on published research on how Architects/Designers do their work.



Two Minds Approach is a theory that software design thinking comprises the following modes of thinking:

- Mind 1 is the *design reasoning mind* with a *problem solving mindset*;
- Mind 2 is the *reflective mind* with a *feedback mindset*

Mind 1 is about design argumentation, whereas Mind 2 is about conscious questioning and reflection on how well we reason and argue with a design.

"Reflective Approach for Software Design Decision Making"
M Razavian, A Tang, R Capilla, P Lago