Beyond the expected.®

PEKIN®

INSURANCE

DevOps Operating Model

DevOps Operating Model Guiding Principles

Technology integrations and service operations connect seamlessly to create a DevOps organization that can show quick successes and agile, rapid responsiveness to customer's application development needs

DevOps First	▪ Extend Agile philosophies by embracing constant testing and delivery through iterative changes ▪ Empower "two pizza teams" to continuously integrate and deliver change ▪ Culture of multiple releases and rapidly deploying code to production
Extreme Automation	▪ Zero touch operations organization ▪ Self service provisioning
Measure Everything	▪ Continuous monitoring and measurement ▪ Metrics driven change
Fail Fast Experimentation	▪ Fail fast, Fail often SDLC cycle ▪ Continuous Feedback loops
Cloud First	▪ IT organization includes cloud as part of business and technology strategy and solution ▪ New development is cloud native applications

DevOps Operating Model Options

The following IT operating models are commonly seen across organizations

Centralized ⟷ Decentralized

Conservative ⟷ Contemporary

	Component Centric	Service Centric	Value Chain Centric	Customer Centric
	IT Components	Services	Plan/Build | Run	Buss. Svcs. IT | Buss. Svcs. IT | Buss. Svcs. IT | Buss. Svcs. IT | Buss. Svcs. IT
	IT Business Management	IT Business Management	IT Business Management	
	Program Management	Program Management	Program Management	
	Infrastructure & Shared Services	Infrastructure & Shared Services	Infrastructure & Shared Services	Infrastructure & Shared Services
Description	▪ Organize by IT component ▪ Within each component, responsible for all elements of value chain	▪ Organize by IT services (e.g., Application Hosting, End User Solutions, Security, Messaging)	▪ Organize by IT Value Chain ▪ Organize by combination of Plan/Build/Run	▪ Dedicated IT infrastructure set of services (including Project Management) delivered to internal customers
When the model is applicable	▪ Need to manage all resources within defined IT Service Components ▪ Resource constraints across value chain ▪ Need for increased responsiveness	▪ A mature organization with high degree of experience in executing the delivery of "bundled offerings" ▪ Service catalogs and service level agreements are defined ▪ Chargebacks are based on consumption of service	▪ Effective with a clear enterprise IT platform strategy ▪ Need for separation of value chain and for increased accountability and monitoring of business value ▪ Useful as a foundational model for transformative IT organizations	▪ Highly differentiated set of IT business requirements ▪ Shared Service models are for core services only (e.g., help desk, e-mail, etc.) ▪ Different and changing needs across Business Units, Functions, and/or geographies
Pros	▪ Increased standardization ▪ Clear responsibility and accountability	▪ Higher degree of flexibility in decision-making ▪ Business units are able to operate with fewer dependencies	▪ Centralized structure for establishing direction ▪ Enables evolution of new services into and from the IT organization ▪ Pushes down decision making	▪ Empowers decentralized decision-making and increases responsiveness ▪ Aligns with core concepts of multi-speed IT ▪ Relatively quick to establish
Cons	▪ No coordination between business units ▪ Encourages silos ▪ Encourages a one size fits all approach	▪ Formal separation of business and technical architectural roles ▪ Requires high degree of maturity and experience	▪ Reduces flexibility ▪ Not fully aligned to multi-speed IT ▪ Too many layers	▪ Consensus generally takes longer ▪ Challenging coordination ▪ Issues around co-owners of data and responsibility
e.g.	Wells Fargo, Apollo Group	CVSHealth	Discover, American Express, Capital One	amazon, Google, NETFLIX

Deloitte.

Sample organizations for each operating model are based on external research only and may not reflect current structures

PEKIN INSURANCE

Current State

Pekin's current state IT Operating model is a combination of the component centric and value chain centric model

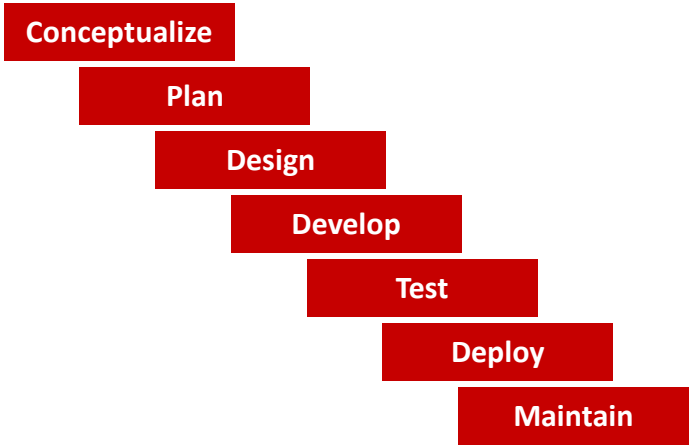
Key Features:

- Leadership is horizontal and ownership is at the application/ component level (e.g., legacy, distributed systems)
- Approach to decision making is top down in most cases, participation from all stakeholders in initial planning is limited
- Individual teams are structured around the delivery value chain. Team responsibilities are divided between plan and building (i.e., BA, developers, QA) and run (i.e., deployment, database and infrastructure)
- The CoE supports project management for large and cross functional projects
- Solution architecture responsibilities are fulfilled by application SMEs or by developers
- EARB reviews and approves architecture design and implementation at defined project stages

	Component Centric	Value Chain Centric	
	IT Components	Plan/Build	Run
	IT Business Management	IT Business Management	
	Program Management	Program Management	
	Infrastructure & Shared Services	Infrastructure & Shared Services	
Description	▪ Organize by IT component ▪ Within each component, responsible for all elements of value chain	▪ Organize by IT Value Chain ▪ Organize by combination of Plan/Build/Run	
When the model is applicable	▪ Need to manage all resources within defined IT Service Components ▪ Resource constraints across value chain ▪ Need for increased responsiveness	▪ Effective with a clear enterprise IT platform strategy ▪ Need for separation of value chain and for increased accountability and monitoring of business value ▪ Useful as a foundational model for transformative IT organizations	
Pros	▪ Increased standardization ▪ Clear responsibility and accountability	▪ Centralized structure for establishing direction ▪ Enables evolution of new services into and from the IT organization ▪ Pushes down decision making	
Cons	▪ No coordination between business units ▪ Encourages silos ▪ Encourages a one size fits all approach	▪ Reduces flexibility ▪ Not fully aligned to multi-speed IT ▪ Too many layers	

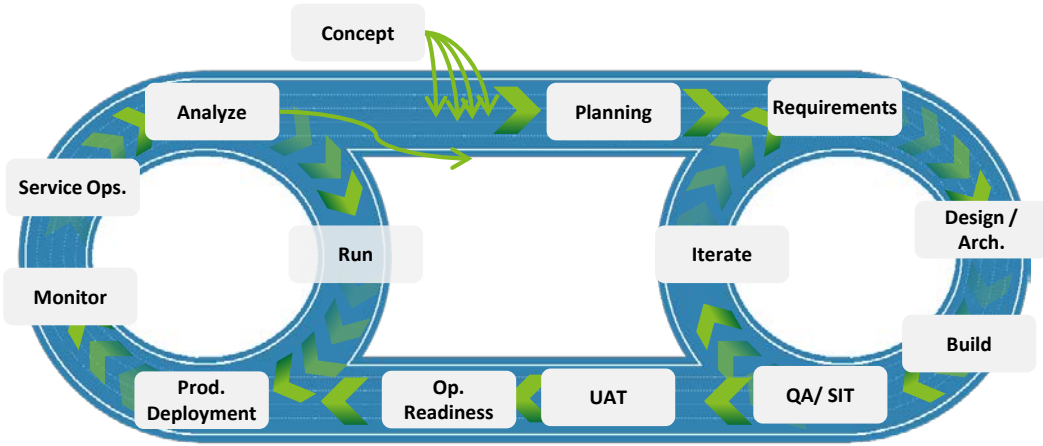
Modernization of the SDLC Framework

Current



Conceptualization to size project scope and effort and to determine resources and teams involved (e.g., architect, data, infrastructure)	Goal is to have scope freeze 1 month prior to release cycle
Gather and refine requirements	2-3 weeks/ varies with project complexity
Develop code to fulfill requirements	6 weeks
SIT to test developed functionality Configure testing environments Conduct regression testing	
Conduct user acceptance testing	After completion of SIT
Product deployed in production	1 week
Feature used by end customer	**3- 6 months**

Target



Conceptualization (due to business needs, regulatory compliance, new ideas, innovation) feeds the release planning process (1-2 meetings)	1 month prior to release planning	Quarterly
Plan, identify, prioritize and groom the features (1-2 meetings)	2 weeks prior to release planning	Quarterly
Release planning - Feature requirements are documented for each sprint and broken down into stories for development	3 days prior to sprint 1 start	Quarterly
Refine and groom user story for the upcoming sprint	2 days prior to sprint start	Each sprint
Development, build , and QA/ SIT go through multiple iteration to develop a feature (Each sprint)	4 weeks	Each sprint
UAT determines readiness for use	Based on completed features	When ready
Feature deployed in production and is used by end customers	**Anytime within 3 months**	**When ready**

Proposed Customer and Product Centric IT

Defining a Product

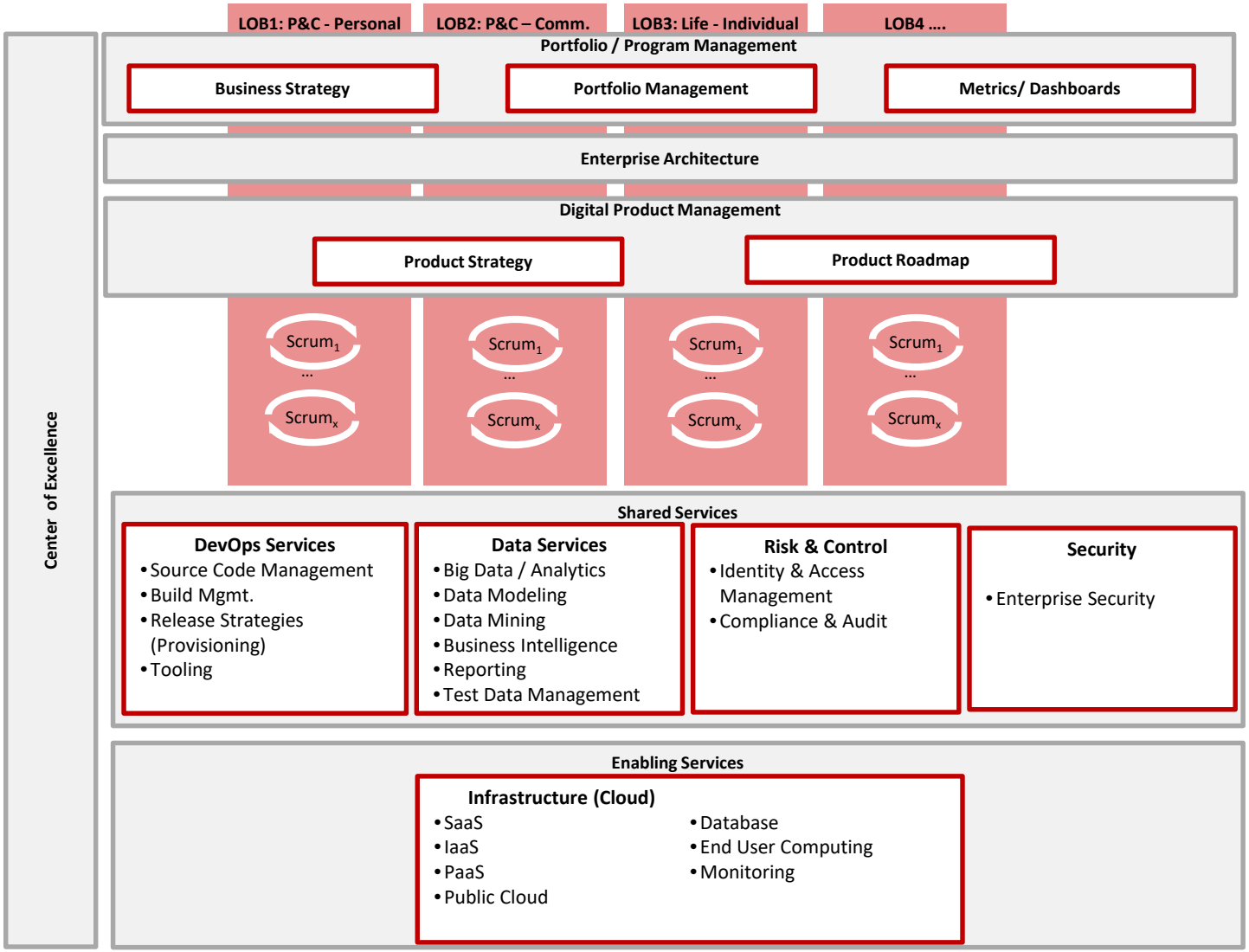
- A product transforms a market opportunity into a service or product available for sale, it may be tangible or intangible
- Product development may involve modification of existing product features, addition of new features or formulation of an entirely new product that satisfies a newly defined customer want or market niche
- Product development phases are conceptualization, design, development, marketing and management

Pekin Product View

P&C - Personal	P&C - Commercial	Life

Proposed Operating Model – Key Areas

This operating model represents an organization with high DevOps maturity, rapidly responding to changing market needs



Proposed Operating Model Changes

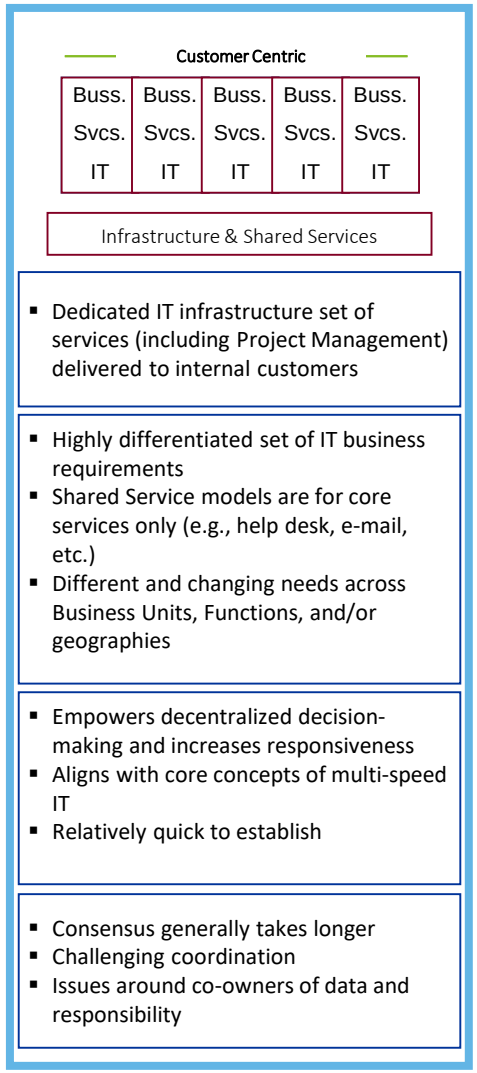
Based on Pekin's desired positioning in the marketplace, we propose a customer-centric IT Operating Model as the foundation for the Business Accelerator

A customer-centric approach is best aligned to Pekin's objectives of delivery with quality and speed:

- Ability to act and make decisions aligned to the business segments to empower customer-centric decision making
- Accountability to develop, deploy and support applications is centered on delivery teams
- Oversight and governance activities are streamlined and applied at enterprise level when necessary

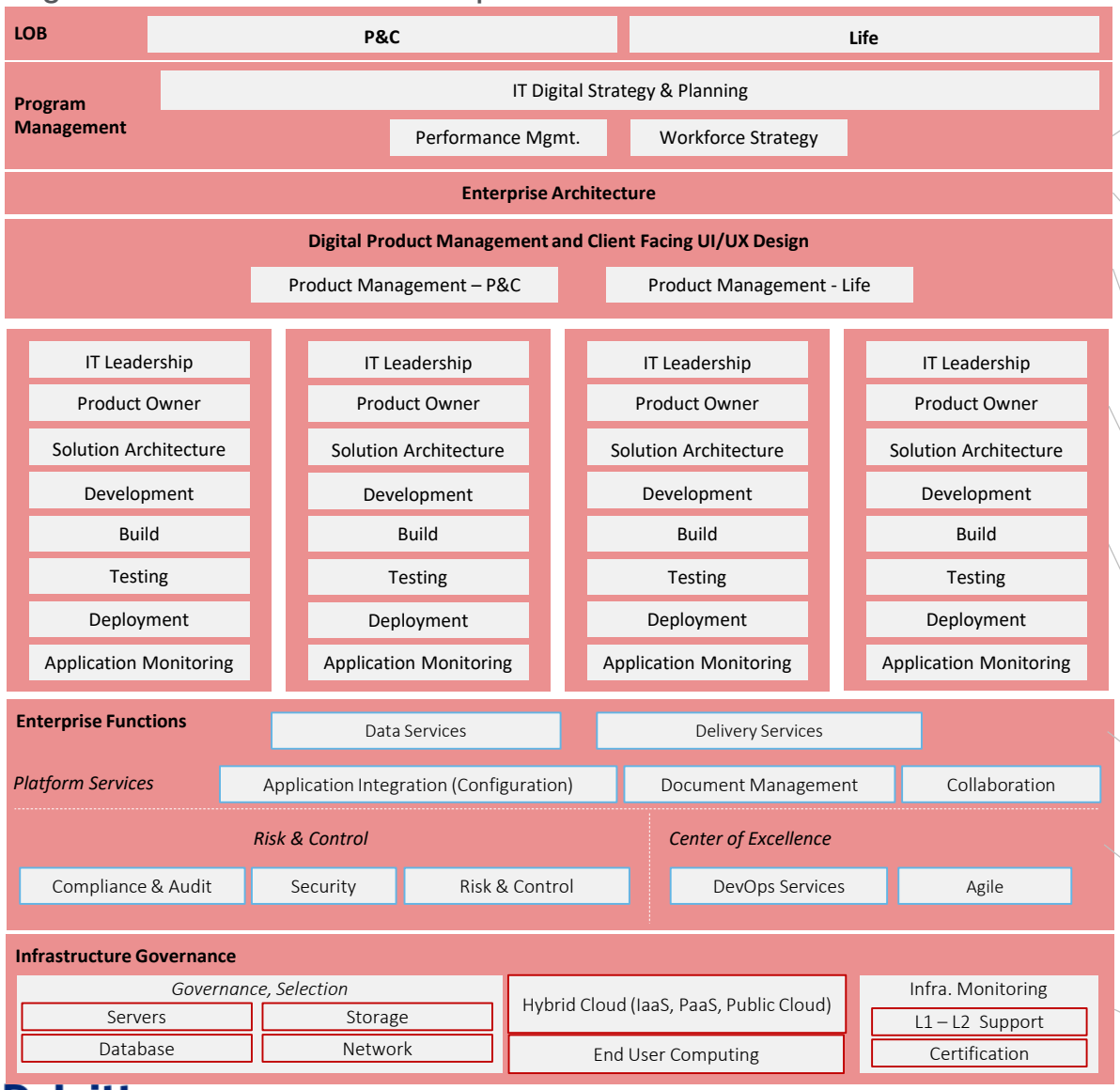
Proposed Changes:

- Teams are structured vertically to support individual lines of business
- Each team should be self contained and should include:
  - Product Owners
  - Solution architects
  - Developers
  - QA/Testers
  - Deployment and release resources
  - Operations
  - Application and infrastructure support group
- Center of Excellence (CoEs) are horizontal and cut across lines of business. CoEs define standards for agile delivery, release management, continuous delivery, cloud, etc.
- Enterprise services are made available for application integration, infrastructure governance (including cloud), security, and other roles



Proposed Operating Model – Function View

The function view of the proposed operating model summarizes high level activities and alignment across the enterprise.



Strategy
The enterprise project management office (PMO) enables realization of Business and IT strategy by prioritizing, budgeting, and planning investments and associated delivery. The group also manages the workforce mix, project metrics and reporting

Enterprise Architecture
Designing, governing, integrating, and managing technology architecture policies, provide guidelines to solution architects. Review project design and implementation through the EARB

Product Management and UI/UX Design
Provide customer/business requirements, align with the LOBs to provide agents/ customers with a consistent and unified user experience

IT Leadership
Manage and analyze the intake/ demand from the business, triage, assign work, and ownership of end to end application development

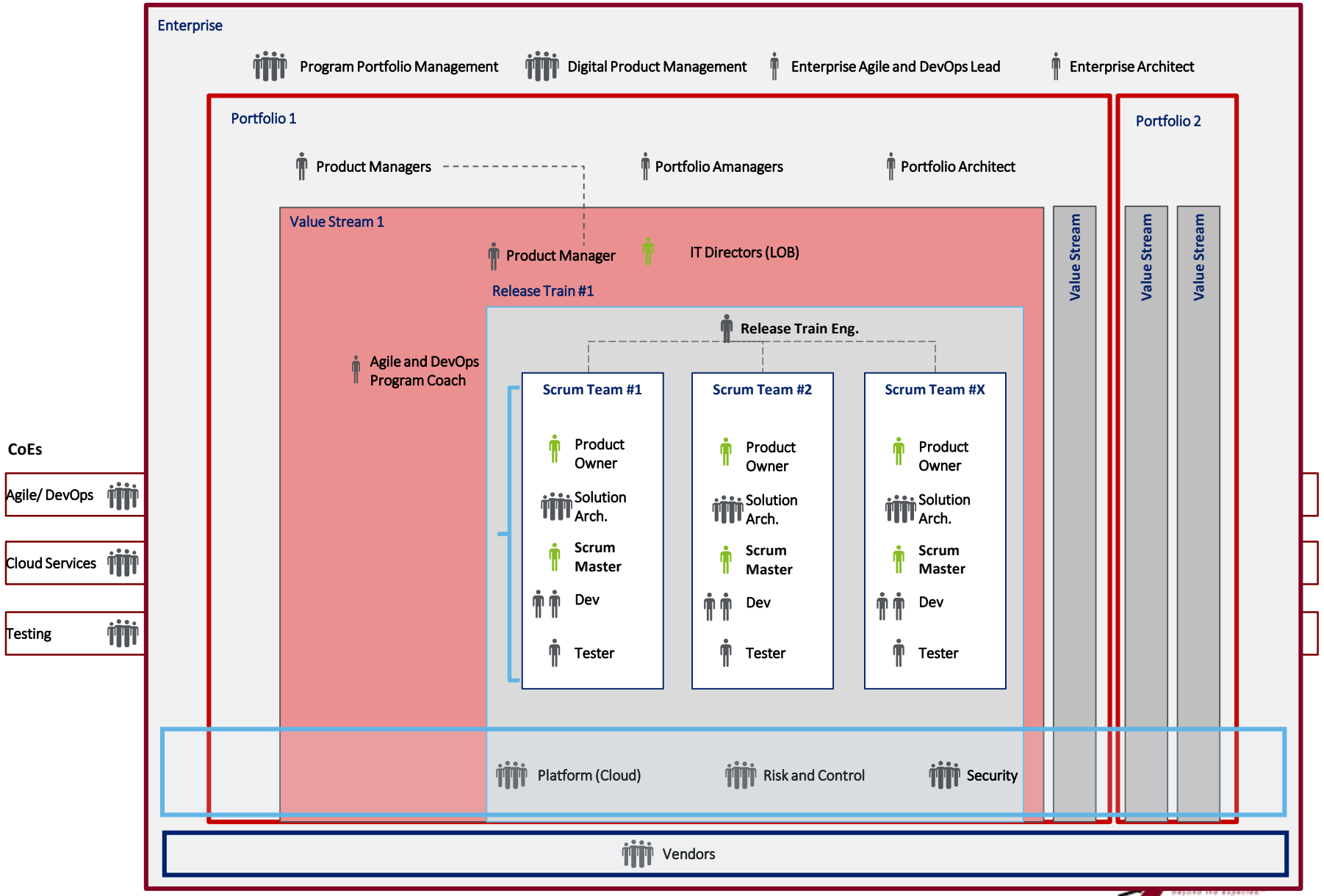
IT Delivery
Liaison with business to identify and analyze business requirements. Prioritize and translate requirements into solutions including associated development, build, test, deploy, release and run activities

Enterprise Functions
Provide streamlined enterprise functions and support for a successful product development

Centers of Excellence
Provide governance over standards, methods, tools, and performance metrics; confirm all teams are delivering according to best practice

Infrastructure
Provide cloud and infrastructure services to enable delivery teams to create and use environments for dev, test, and production

Proposed Operating Model – Role View



Proposed Operating Model – Roles Defined (1/2)

The roles identified to fulfill the target operating model are defined below

Level	Role	Role Description
Enterprise	Project Portfolio Management	Portfolio Managers have primary responsibility for strategy and investment funding, program mgmt., and IT and business governance within a line of business. Act as epic owners
	Enterprise Agile Coach	Leads the agile CoE that defines processes and standards for agile delivery and coaches the program agile coaches
	Enterprise DevOps Coach	Leads the DevOps CoE that defines processes and tools for continuous delivery (development, build, test, deploy) and coaches the program DevOps coaches
	Enterprise Architect	Enterprise architects maintain a high-level, holistic vision of enterprise solutions and development initiatives for a given segment and provide guidance to solution architects and owns the EARB
Portfolio	Product Manager	Product manager are responsible for defining and communicating the program vision to the scrum teams, they are aligned with and provide guidance to release trains
	Portfolio Architect	Portfolio architects coordinate with product managers to maintain a high-level, holistic vision and architecture of the solution
	Infrastructure	Infrastructure provisions, configures and manages infrastructure components (compute, storage, networks and includes cloud)
	Platform Services	Platform services and tools provide application integration and collaboration
	Risk and Control	Risk and control organizations understand the risks the organization is exposed to, and put controls in place to counter threats
	Security	Security defines and enforces enterprise wide policies for protecting infrastructure, information assets, customer data, financial information and other critical IT information
Value Stream	IT Director	The director is accountable for end to end IT delivery and operations specific to a value stream
Release Train	Product Manager	Product manager are responsible for defining and communicating the program vision and backlog. Product managers participate in UAT.
	Release Train Engineer	RTE facilitates program level processes and execution, escalates impediments, and drives release train to continuous improvement

Proposed Operating Model – Roles Defined (2/2)

The roles identified to fulfill the target operating model are defined below

Level	Role	Role Description
VS, ART, scrum team	Program Agile Coach	An Agile coach identifies and coaches on Agile best practices and principles, contributes to the agile CoE and aligns on standards
	Program DevOps Coach	A DevOps coach identifies and coaches teams on continuous delivery processes and tools, contributes to CoE and aligns on standards
Scrum Team	Product Owner	The Product Owner has the vision of what a scrum team intends to build and communicates that vision to team. The PO performs analysis and interfaces with the business.
	Solution Architect	SAs have the technical responsibility for the overall architecture of the solution or an application and ensure it aligns with the enterprise architecture
	Scrum Master	The Scrum Master is responsible for conducting the ceremonies of the scrum team and works to remove any impediments to the development efforts
	Developers	Developers work with the shared services and infrastructure to develop functionality in accordance with the user story. In addition are responsible for monitoring applications in all environments and ensure built in quality
	Testers	Testers create functional, performance, and security test scripts and monitor testing execution as part of the continuous integration process
Center of Excellence (CoE)	Agile	The agile CoE defines processes and standards, ensures agile maturity and defines measures of success
	DevOps	The continuous delivery CoE defines processes and owns tools for continuous delivery, ensures DevOps maturity and defines measures of success
	Cloud Services	The Cloud Services CoE provides governance, standards and coaching on the use of cloud as infrastructure and use of cloud services
	Testing	The testing CoE establishes testing/ quality assurance processes and ensures adoption of the processes
	Vendor	A *vendor* is an internal or external organization that develops and delivers components, subsystems or services that help Solution Trains provide Solutions to their customers.

Taxonomy of Work

Portfolio / Program Backlog		Release Backlog	Sprint Backlog
			
Portfolio Epic (Years)	**Program Epic (Features)** (Quarters)	**User Story** (Days)	**Sub-Task** (Hours)
• Long term initiatives that align to the strategic business objectives/ themes • Has measureable target performance metrics to determine realization of business value	• Medium term initiatives that enable the implementation of the Portfolio Epic • Enable iterative delivery of business value desired by the Portfolio Epic • Has measureable target performance metrics to determine realization of business value	• Small, independently valuable work items that make up an Epic and can be delivered by one Tech Product • Will need to meet the Tech Product Team's Definition of Ready (DoR) to be worked on in a sprint	• Activities needed to complete an User Story • May be repetitive "chores" to support or maintain the product
Bounded by: Enterprise/ Portfolio	**Bounded by:** Portfolio Manager	**Bounded by:** Product Owner	**Bounded by:** Team
Owned by: Portfolio Manager/ Product Manager	**Owned by:** Product Manager	**Owned by:** Product Manager	**Owned by:** Product Owner