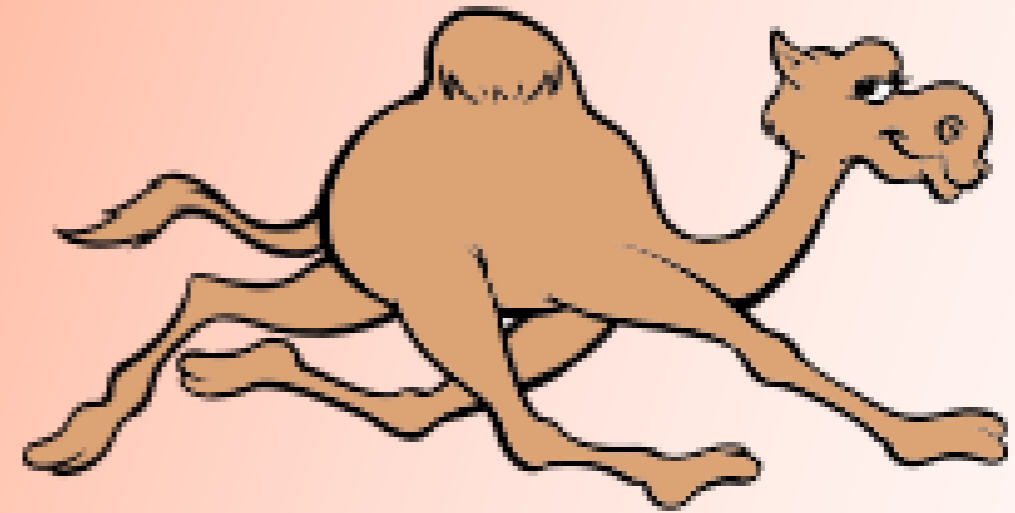# CamelOne 2013

June 10-11 2013
Boston, MA

Introduction to the Integration Technology Selection Guide

By Asheesh Goja , UPS

About Me

Asheesh Goja

- Architect at UPS Enterprise Architecture

- One of the founding architect of the Integration Competency Center at UPS

- Developed a perti-net based DSL for Application Integration

Agenda

- UPS background

- Integration challenge

- UPS ICC

- Technology selection guide

  - Approach

  - Process

  - Solution Patterns

  - Decision Matrix

- Summary

- Questions

CamelOne 2013

UPS

The world's largest package delivery company

Scale

- Global leader in logistics and supply chain management

- 9th largest airline

- One of the top 10 global freight forwarding companies

- One of the world's largest customs brokers

- Moves 6% of US and 2% of world GDP

- Operates in 220 countries and territories

- Delivers more than 16 million packages a day to 8.8 million customers

Scope

- Small package

- Air and Ocean Freight

- Less Than Truck Load (LTL) and TL

- Customs Brokerage

- Retail shipping

- Health Care Logistics

- Critical Parts Management

- Warehousing

- Assembly / Repair Service

- Store Automation

- Finance

Technology

- 2 data centers

- 10 Mainframes

- 72,979 MIPS capacity

- 16.1 PB of storage

- 194,483 Laptops and Workstations

- 18,230 Physical Servers

- 108,899 DIADS in daily use

- 3,311 Network Sites

- 26.8 M page views per day
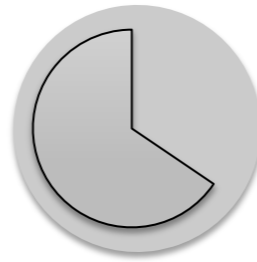
- 39.5 M tracking request per day
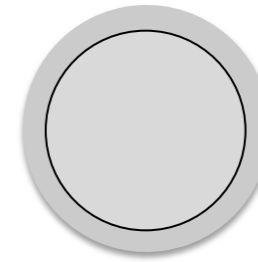
The Integration Challenge
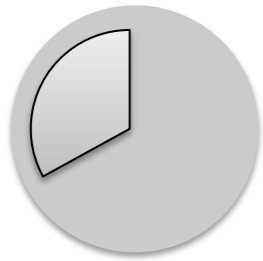
Technology

- Multiple incompatible technologies
- Proliferation of integration technologies
- Independently developed applications and their inter-connections

Process

- Duplication of work – redeveloping interfaces multiple times
- Underestimating integration effort and costs
- Low short term cost of point-integrations
- Integration is not limited to just A2A, need to address B2B, Data etc.

Organization

- Multiple lines of business
- Varied applications resulting from acquisitions
- Existing integration vendors

Addressing The Challenge

> *Integration is fundamentally different from software application development - It is deployment-centric, not development-centric*

Took a holistic approach that address technology, process and people

Formed a competency center dedicated to technology, process, governance and direction

- Clarified roles and responsibilities

- Developed common procedures and documented best practices

- Defined metadata standards and canonical models

- Provides training and "onboarding" services

- Created an integration requirements questionnaire

- Developed an integration technology selection guide

CamelOne 2013

The Selection Guide – Drivers

- Multiple integration architectures, technologies and platforms

- Selection, usage and standardization is inherently complex with limited industry guidance/consensus

- And hence the need for a selection guide that
    - Disambiguates integration terms, concepts and definitions
    - Assists in identifying integration scenarios
    - Delineates recurring design problems
    - Maps them to solution patterns
    - Presents the guidance as an decision tree
    - Enables evaluating integration products on implemented solution patterns rather than vendor advertised feature sets.
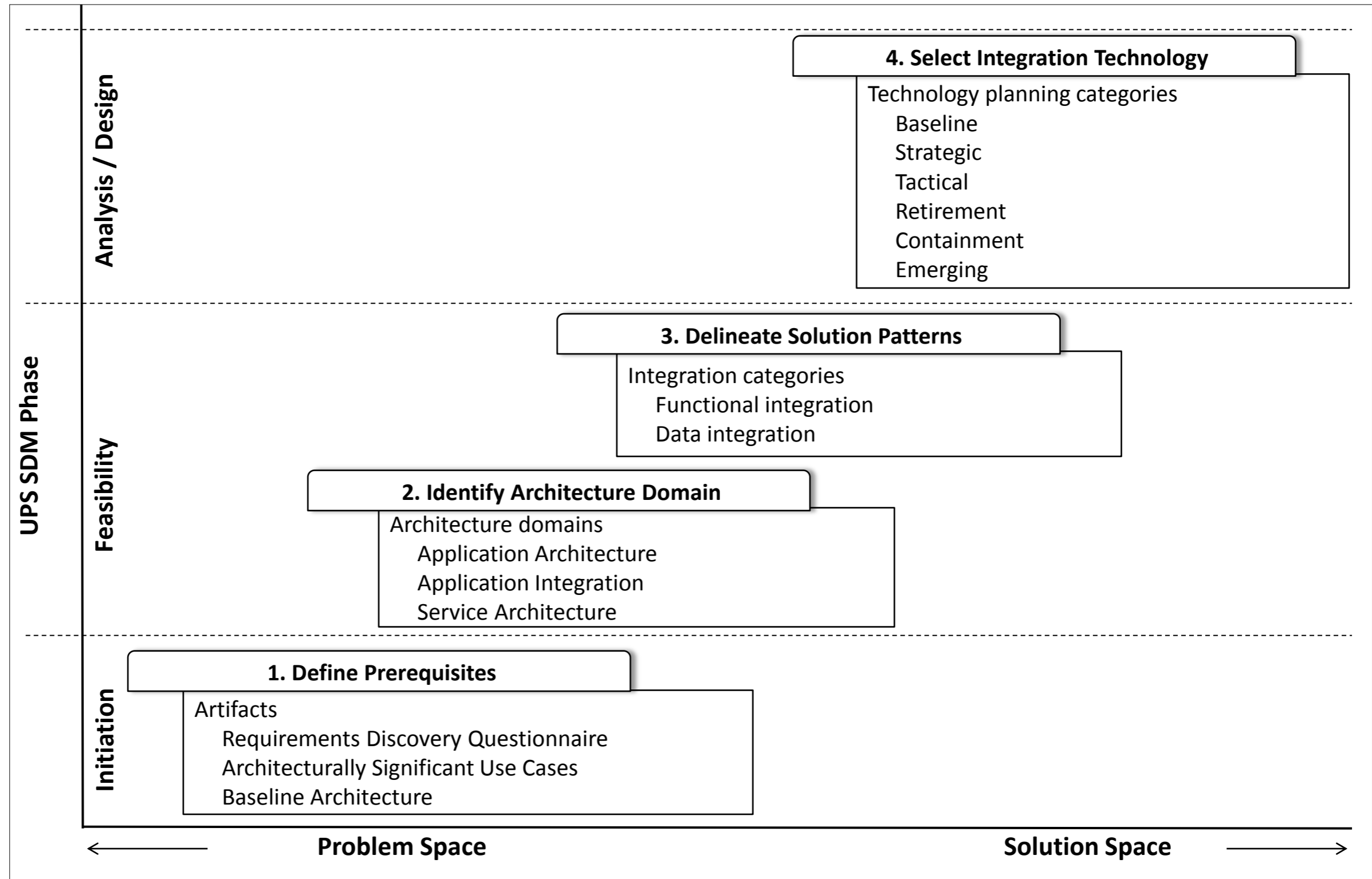
CamelOne

The Selection Guide - Approach

- Examine integration scenarios

- Identify integration architecture domains and recurring integration problems

- Find reusable solutions to these problems

- Express the solutions as technology agnostic patterns

- Use these patterns to create a selection decision matrix.

- Formalize this approach as a process that is aligned with the UPS solution development methodology (SDM)
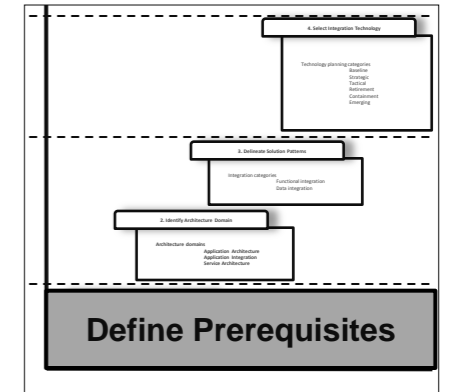
CamelOne 2013

Four Step Selection Process

UPS SDM Phase

Analysis / Design

4. Select Integration Technology

Technology planning categories
- Baseline
- Strategic
- Tactical
- Retirement
- Containment
- Emerging

Feasibility

3. Delineate Solution Patterns

Integration categories
- Functional integration
- Data integration

2. Identify Architecture Domain

Architecture domains
- Application Architecture
- Application Integration
- Service Architecture

Initiation

1. Define Prerequisites

Artifacts
- Requirements Discovery Questionnaire
- Architecturally Significant Use Cases
- Baseline Architecture

← **Problem Space**   **Solution Space** →

CamelOne

Step 1 – Define Prerequisites

Define Prerequisites

- **Integration requirements questionnaire**
  The questionnaire assists in surfacing, identifying and prioritizing integration requirements

- **Architecturally significant use cases**
  Use cases that "exercise" the most critical parts of the application architecture

- **Baseline architecture**
 High level 'as is' or 'to-be' architecture of an application.

Step 2 – Identify Domains

Identify Domains
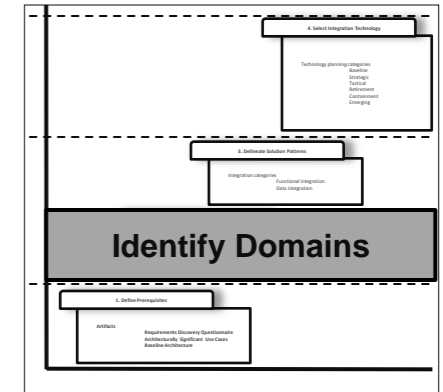
- **Application Architecture**
  The focus of integration in this domain is on integrating core application(s) with add-on application(s)

- **Application Integration**
  The focus of integration in this domain is on integrating independent applications with one another

- **Service Architecture**
  The focus of integration in this domain is on service enabling application(s)

Step 3 – Delineate Patterns

Delineate Patterns
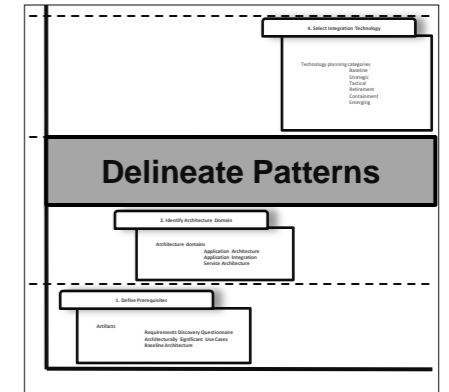
Pattern Classification

- **Functional Integration**
  Patterns in this category enable exposing the functionality for integrating applications while the internal data is encapsulated

- **Data Integration**
  Patterns in this category enable exposing internal data for integrating applications instead of functionality
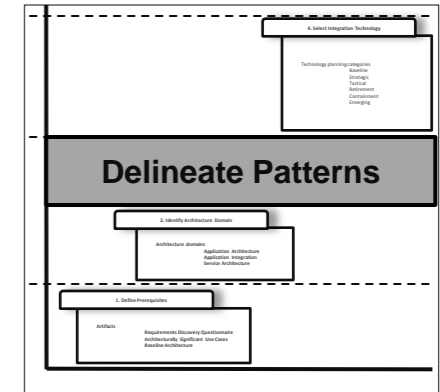
Pattern Structure

- **Problem context**
  Describes the situation or scenario that give rise to this problem

- **Primary use case**
  Describes the recurring integration problem arises in that context

- **Solution**
 Describes a proven and reusable resolution of the problem

Step 3 – Patterns Example

Architecture Domain: Application Integration
Pattern Category : Functional
Pattern Name : Integration Broker

Delineate Patterns

Problem Context	Primary Use Case	Solution– Integration Broker
Business solution spans over multiple heterogeneous distributed applications with varied technical architectures resulting in duplicate integration logic, redundant interfaces, inconsistent data, and un-traceable interactions.	Integrate applications that exhibit the following characteristics: • Logically and Physically dependent • Conflicting QOS requirements • Multiple point-to-point integrations • Disparate technical architectures • Heterogeneous platforms • Low latency interactions • Complex interactions implemented as application logic	**Architecture Style :** POSA 'Broker' Architecture pattern. **Description:** The integration broker acts as a third-party intermediary that facilitates interactions between participating applications. Integration Broker based application addresses the primary use case by providing the following capabilities: • Invocation • Mediation • Messaging • Protocol Bridging • Adapters • Validation • Enrichment • Transformation
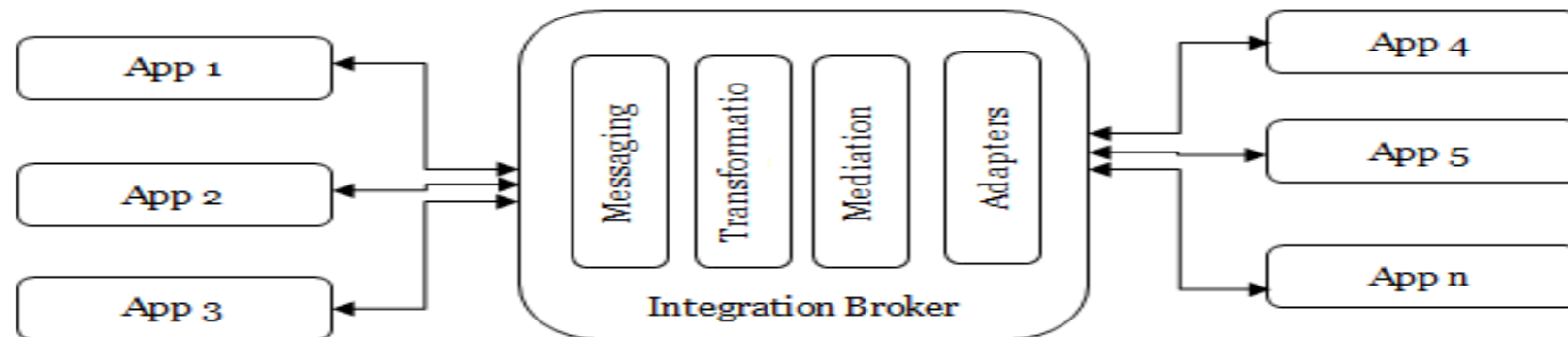
Step 4 – Select Technology
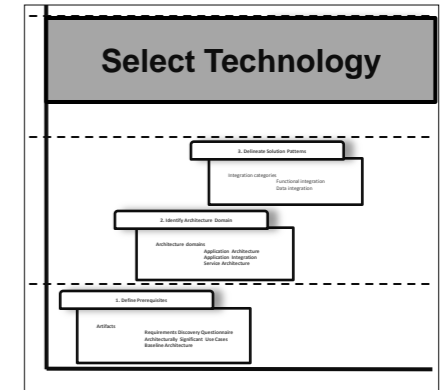

Select Technology

Select a technology implementation of the solution pattern from one of following planning categories:

- **Baseline**
  Technology targeted as the primary deployment/ investment option

- **Tactical**
  Technology used for tactical point solutions

- **Strategic**
  Technology that provides a strategic advantage

- **Containment**
  Technology that is targeted for limited investment

- **Retirement**
  Technology that is targeted for de-investment

- **Emerging**
  Technology that is to be evaluated for future integrations

CamelOne 2013

Step 4 –Technology Example
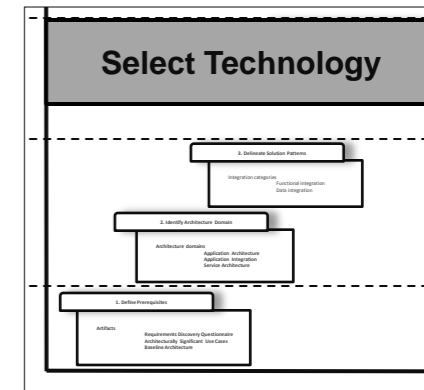
Technology options for the Integration Broker Pattern

Solution Pattern	Baseline	Tactical	Strategic	Containment	Retirement	Emerging
Integration Broker	Vendor Integration Solution	Custom Built App	Vendor CIS Suite	Database Integrator	Legacy Integration App	Cloud Broker

CamelOne 2013

Putting it all together

The Patterns Catalog

Domain	Solution Pattern	
	Functional	**Data**
Application Architecture	• RPC API • Message API • API façade with mediation	• Shared Database • File Transfer • Data Replication
Application Integration	• Process Manager • Integration Broker • Message Broker	• ETL • iPaaS • CDC • MDM • MFT
Service Architecture	• Resource API • Message API • RPC API • ESB • Message Bus • Service Orchestration • Service Choreography	• Entity Aggregation • Data Virtualization • Reference Data Centralization

CamelOne

The Technology Selection Grid

Integration Solution Pattern	Planning Category					
	Baseline	Tactical	Strategic	Containment	Retirement	Emerging
Application Architecture Domain						
RPC API						
Message API						
API façade with mediation						
Shared Database						
File Transfer						
Data Replication						
Integration Architecture Domain						
Process Manager						
Integration Broker						
Message Broker						
ETL						
iPaaS						
CDC						
MDM						
MFT						
Service Architecture Domain						
Resource API						
Message API						
RPC API						
ESB						
Message Bus						
Service Orchestration						
Service Choreography						
Entity Aggregation						
Data Virtualization						
Reference Data Centralization						

CamelOne

Summary

- Integration is challenging

- Technology alone is not enough to address it

- Need to encompass both process and people

- UPS addressed it with a competency center focused on the 'Why' and 'What' of Integration

- A key tool developed for this purpose is the selection guide

- Selection is based on technology agnostic solution patterns

- Solution patterns are overlaid against planning categories to form decision trees

- This enables appropriate selection of integration technologies

CamelOne 2013