SOA UNIVERSE
SOA Knowledge in Action

HURWITZ & ASSOCIATES

liquidhub

Executive Briefing:
Service Oriented Architecture

Prepared by:
Judith Hurwitz, Hurwitz and Associates
Thomas J. Cozzolino – Director, LiquidHub, Inc.

Today’s Agenda

- Introduction to SOA (30 min.)
- Business Architecture of SOA (30 min.)
- Technical Architecture of SOA (45 min)
- Getting Started (45 min.)
- Thoughts on Operationalizing SOA (45 min.)
- Q&A (15 min.)

SOA UNIVERSE

SOA Knowledge in Action

HURWITZ & ASSOCIATES

liquidhub

Section I: Introduction to SOA

Introduction to SOA: 30 min.

- What is SOA?
- Business Perspective
- Why do we Care?
- Discussion Questions

Enterprises face both Business and IT Challenges

- **Your Business Imperatives**
  - Drive new business models and direction
  - Improve business performance and ROI while reducing costs
  - Shorten time to market
  - Minimize risk associated with change
  - Enable mergers, acquisitions, and divestitures
  - Foster customer intimacy with frictionless on-demand business information

Most Enterprise IT Architectures

- 80% of budget spent on maintenance
- Chaotic systems architecture
- Redundant systems
- Increasingly challenging systems integration
- Disconnected from the Business

+

- **Your IT Mandates**
  - Truly link business and IT
  - Reduce costs and complexity, ensure stability and flexibility
  - Optimize assets today and tomorrow
  - Extend value and reach of the Enterprise
  - Consider the right mix of strategic initiatives while meeting the tactical needs of the business
  - Adopt a rational portfolio of applications, not a single packaged application or a technology platform
  - Invest in people, strategies and technologies that enable nimbleness

IT services are under pressure to provide both ***high flexibility*** and ***predictable quality and costs***

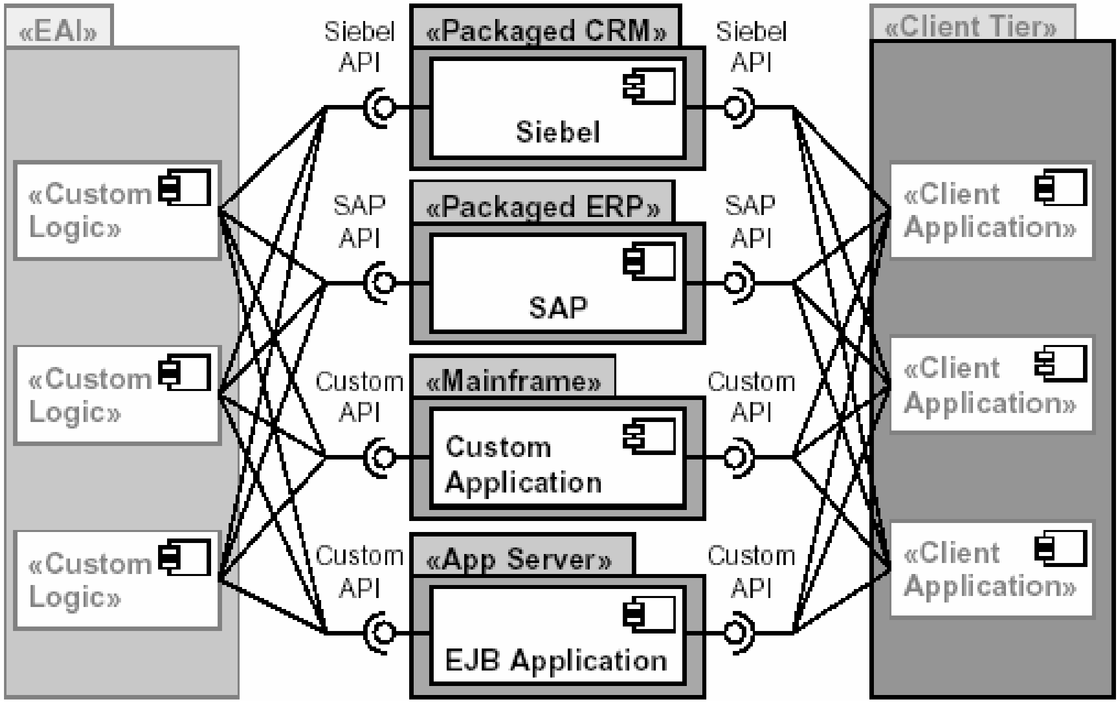
Current Systems Development Challenges

- **Application Development and Integration Issues**
  - Lack of Flexibility
  - Not Standards Based
  - Project Costs and Long Duration
  - Socio-Technical Complexity (Subject Matter and Technical Expertise)

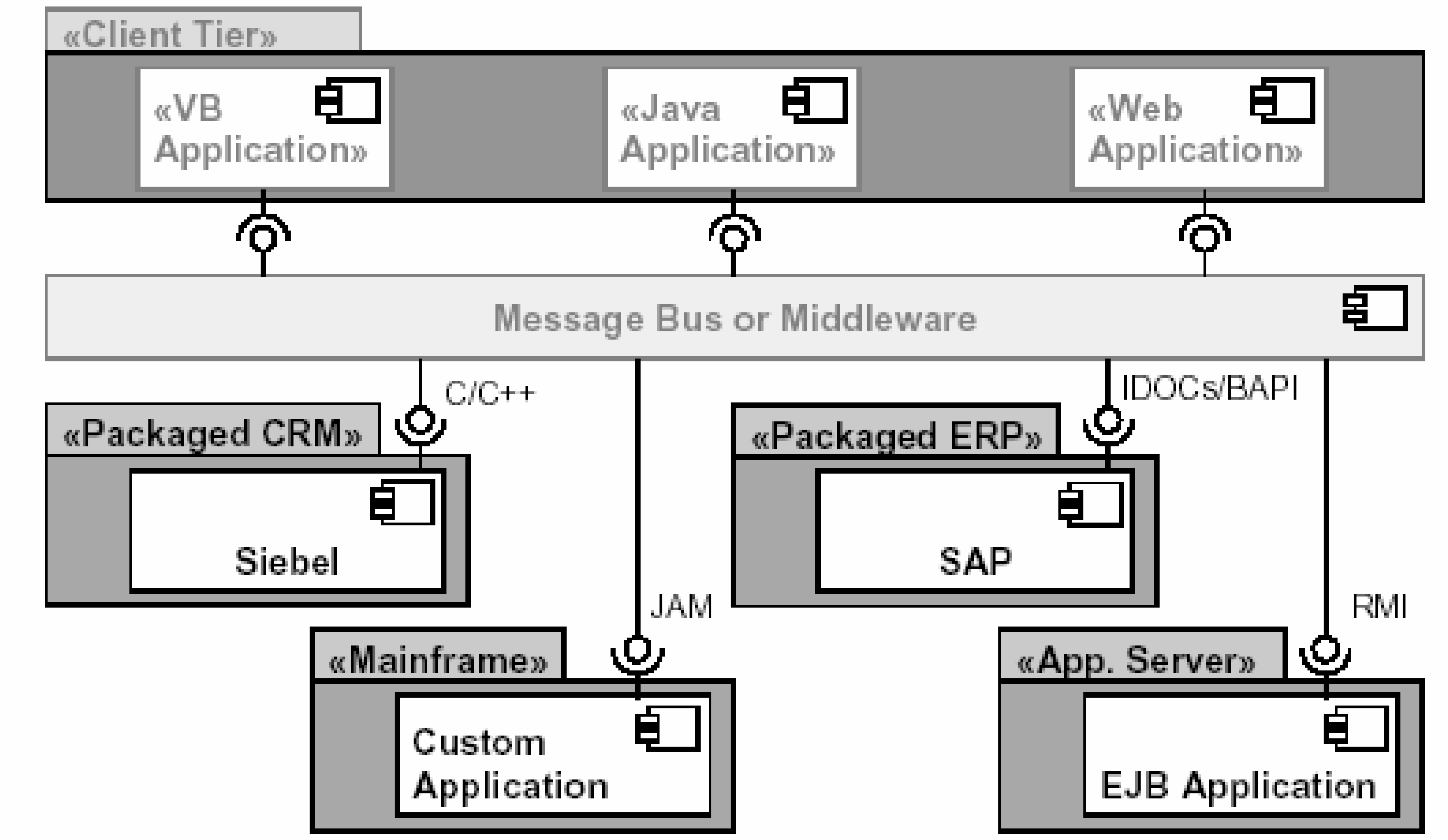
- **Traditional Software Development and Integration Methodologies**
  - Waterfall ☺
  - Agile
    - Extreme Programming (XP)
    - SCRUM
    - Crystal
    - Rational Unified Process (RUP)
  - Point-to-Point (P2PI)
  - Enterprise Message Bus / Middleware (EAI)
  - Business Process Based Integration (BPM & BPR)

Point-to-Point Integration

- Proprietary messages, APIs
- Every link is custom integration
- Duplication of efforts
- Lack of open standards
- Tight coupling of data and implementation
- Skill set issue
- Projects lasting months
- Cost (skill, time, products)
- Operational polices are embedded in application
- Lack of agility
- Slow response by IT to business changes



Historical EAI and BPM



The challenge of accessing, integrating and transforming data (enterprise information integration) has largely been left to developers to perform using manual coding.

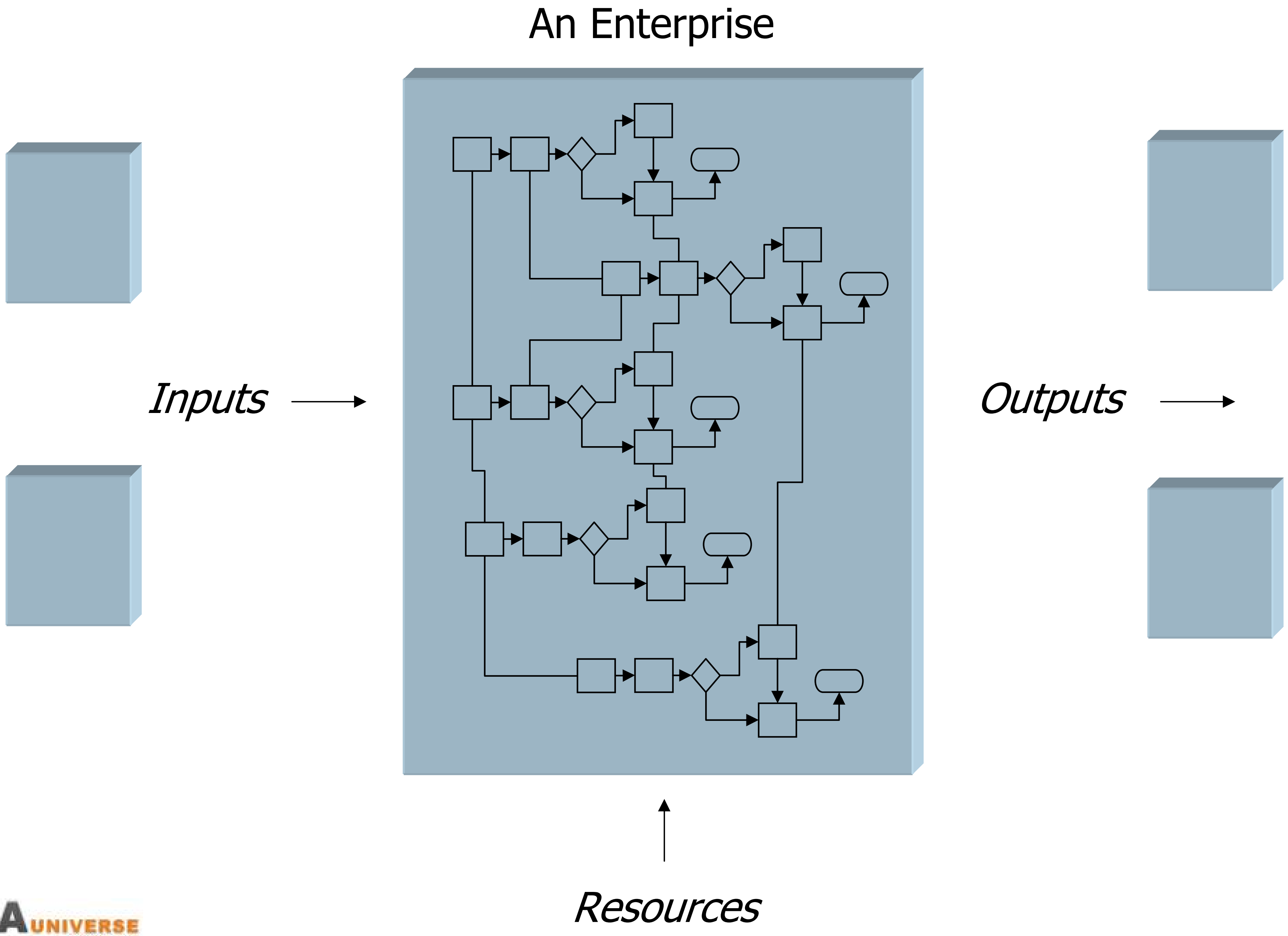
Drivers for SOA: Why Bother?

Business / IT Imperatives

- Business / IT alignment - IT as a Business Strategy
- Concept of an Agile Business, powered by Agile IT

- View of the Enterprise as a set of business processes and capabilities
- Need to reduce costs, trading “friction” and business complexity
- Need for stability, flexibility, scalability
- Need to maximize leverage and use of current IT assets and future IT investments
- Need to extend value and reach of the Enterprise (e.g., to business partners)
- Need to invest in a diversified, vendor-agnostic portfolio of applications

The Enterprise as a Collection of Processes

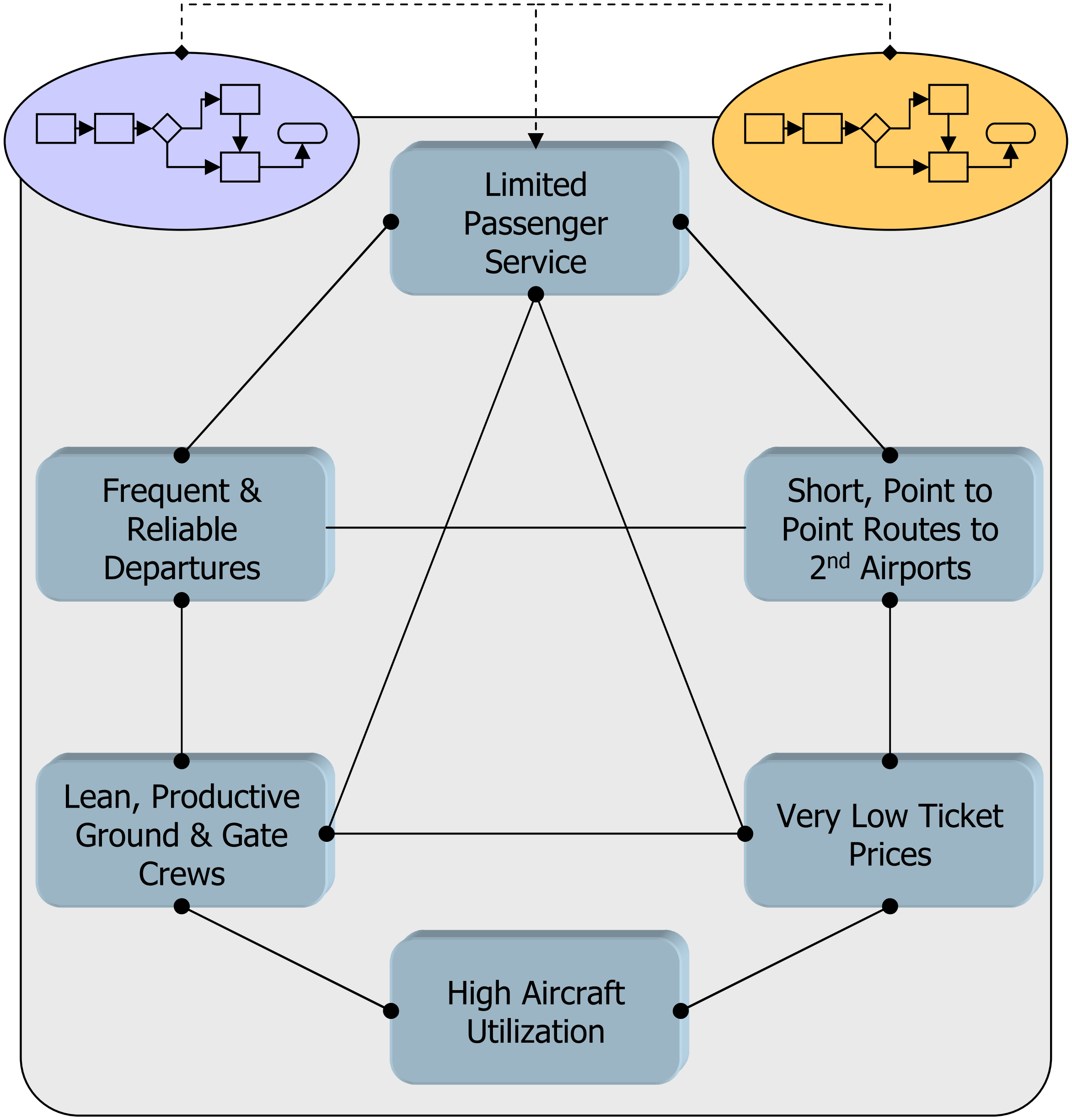


Business Strategy is all about Processes

- Michael E. Porter's influential work has established the importance of thinking about strategy and the role of processes in achieving competitive advantage.

- Competitors, Porter argued, would always try to copy your innovations and "best practices." What they couldn't easily copy was a well integrated Value Chain in which every activity fit together to achieve a well thought out strategy: "**The essence of strategy is choosing to perform activities differently than rivals do.**"

- Senior executives should think in terms of processes - one strategic goal of the organization should be to create value chains and **processes that are unique and that fit together** to give the organization a clear competitive advantage that is difficult for rivals to duplicate.

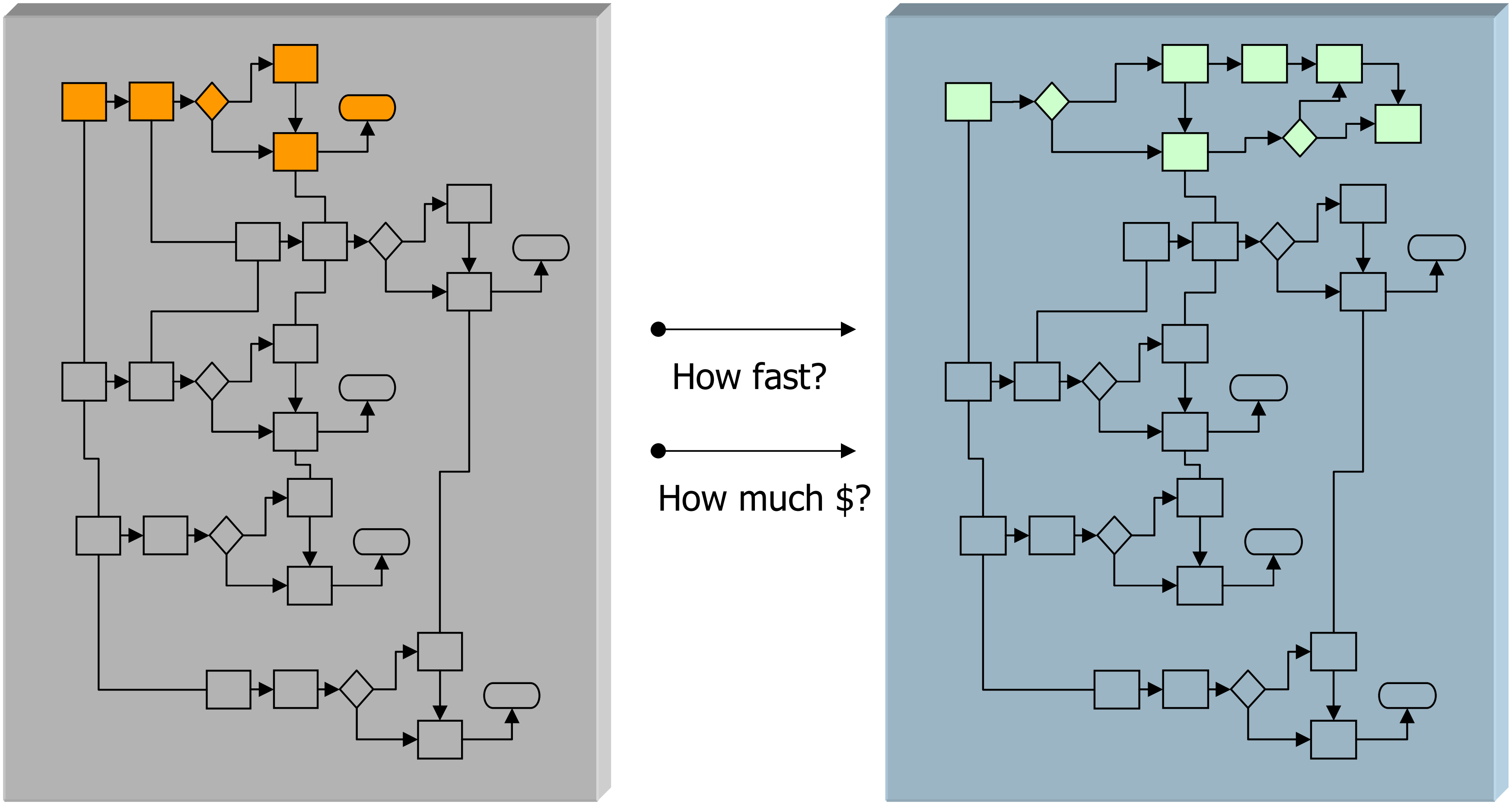

Limited Passenger Service
Frequent & Reliable Departures
Short, Point to Point Routes to 2nd Airports
Lean, Productive Ground & Gate Crews
Very Low Ticket Prices
High Aircraft Utilization

Adapted from "What is Strategy," HBR, Porter 1996

But, what about the Capability to *Change* and/or *Introduce* processes?

Enterprise A

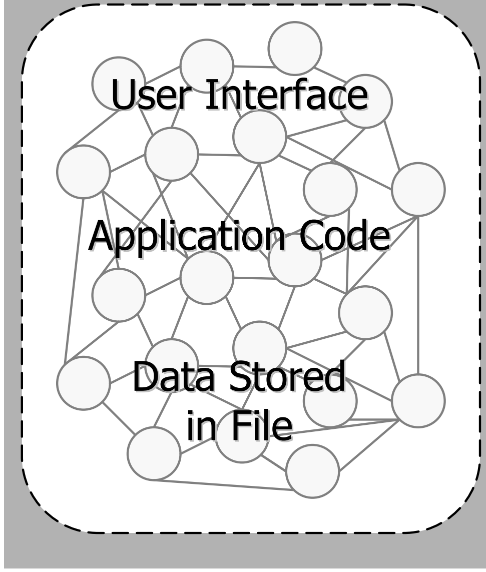
Enterprise A′



The ability to reconfigure Enterprise processes is the essence of 'Agility.'

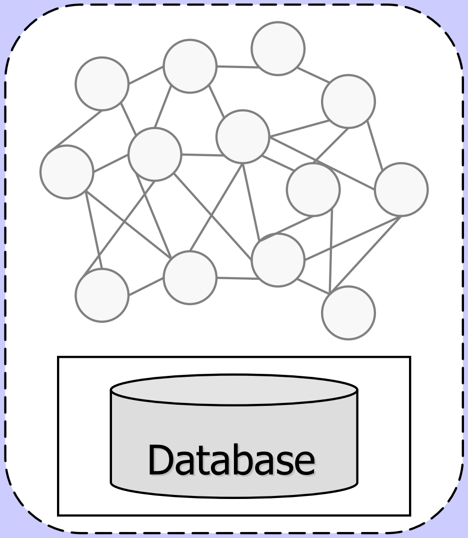
Evolution of Traditional Architectures thus Far…

Phase 1: Monolithic Application Architecture



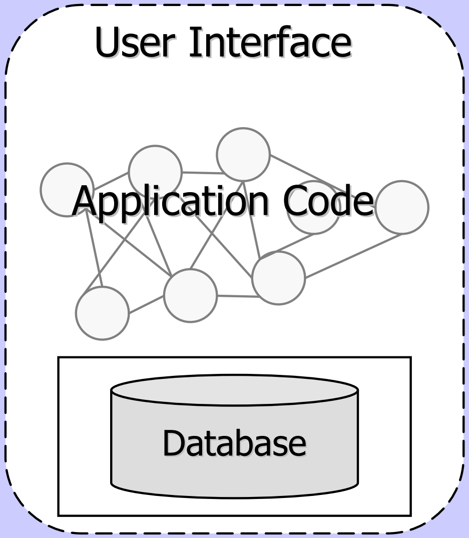
- Early mainframe applications were monolithic, everything was managed by one huge program with no separation of layers.
- The monolithic program handled the user interface, the application code and access to data which were stored in files.
- Hard to adapt & improve; layers are tightly connected; small change can break the entire application.

Phase 2: 2-Tier Application Architecture



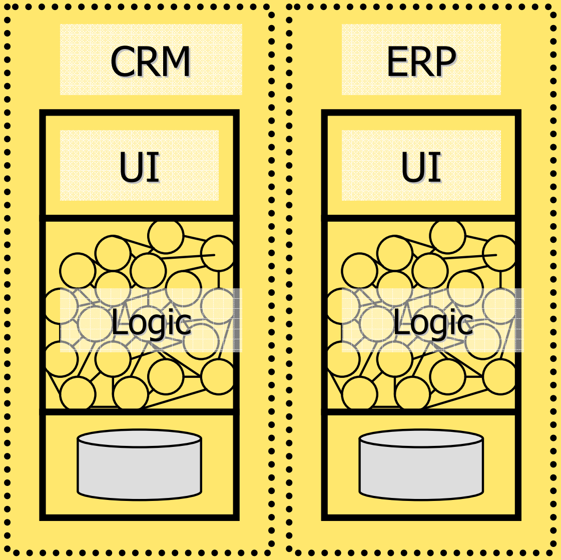
- Once the database was invented, the data was separated into a different layer. In 2-tier mainframe applications, the monolith ran on the mainframe.
- In 2-tier client/server, the monolith ran on a personal computer.

Phase 3: 3-Tier Application Architecture



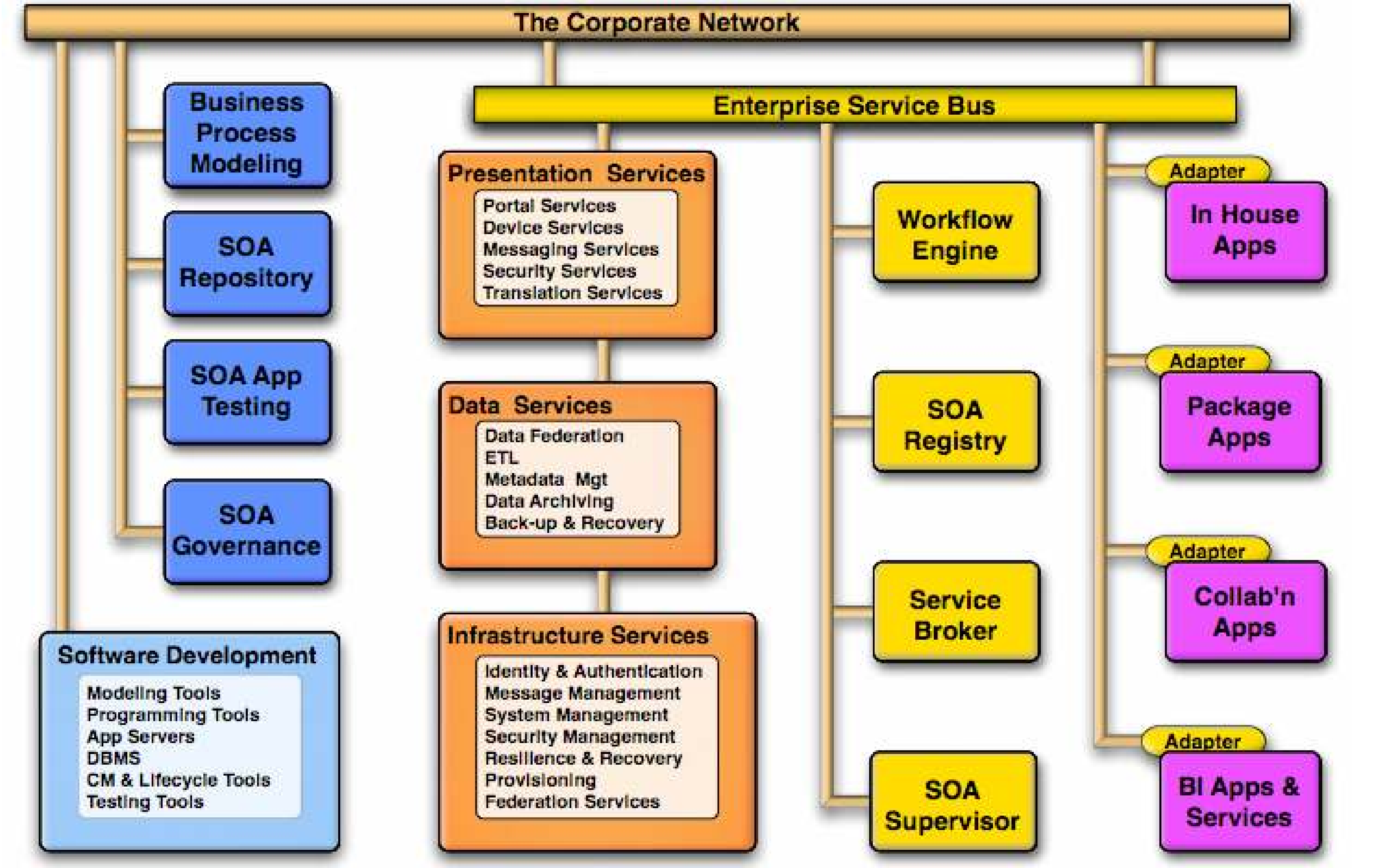
- In the late 1980s & early 1990s, with the help of the PC and network, the 3-tier architecture arrived
- Both the UI and application code would sometimes run on a PC, and sometimes only the UI would run there.
- After the arrival of the Internet, everything ran on the server and the browser became the interface

Phase 4: Distributed Application Architecture



- Collections of 3-tier applications that use either PC apps or browser as the UI
- Applications are silos and UIs of these apps are still bound to the monolith
- Companies reuse portions of the monolith through portals and use APIs to integrate monoliths

…Has Led Us to SOA



SOA UNIVERSE SOA Knowledge in Action

Source: SOA for Dummies/Hurwitz Associates

What is SOA?: A Contemporary Approach to Enterprise Architecture

Service Oriented Architecture can be defined as: “An Architecture for building business applications as a set of loosely-coupled black-box components orchestrated to deliver a well-defined level of service by linking together business processes.*”

SOA-based approaches are effective catalysts for providing Enterprises with new ways to approach current problems:

- **Loosely-coupled black-box components:** Historical point-to-point and other integration strategies are expensive, error-prone, and difficult to manage and scale. Loose coupling drives many benefits, including reuse, predictability, and uniformity.
- **Well-defined level of service:** True SOA-based approaches drive both business and IT to clearly consider, define, and measure Service Levels, greatly increasing the effectiveness of development and providing clear, transparent success criteria.
- **Business Processes:** Critical to success of the Business-IT partnership is the ability to focus on a common set of linkage points. Processes in SOA, long the domain of the Business, finally become the focus area of IT through the construct of Business Process Management (BPM).

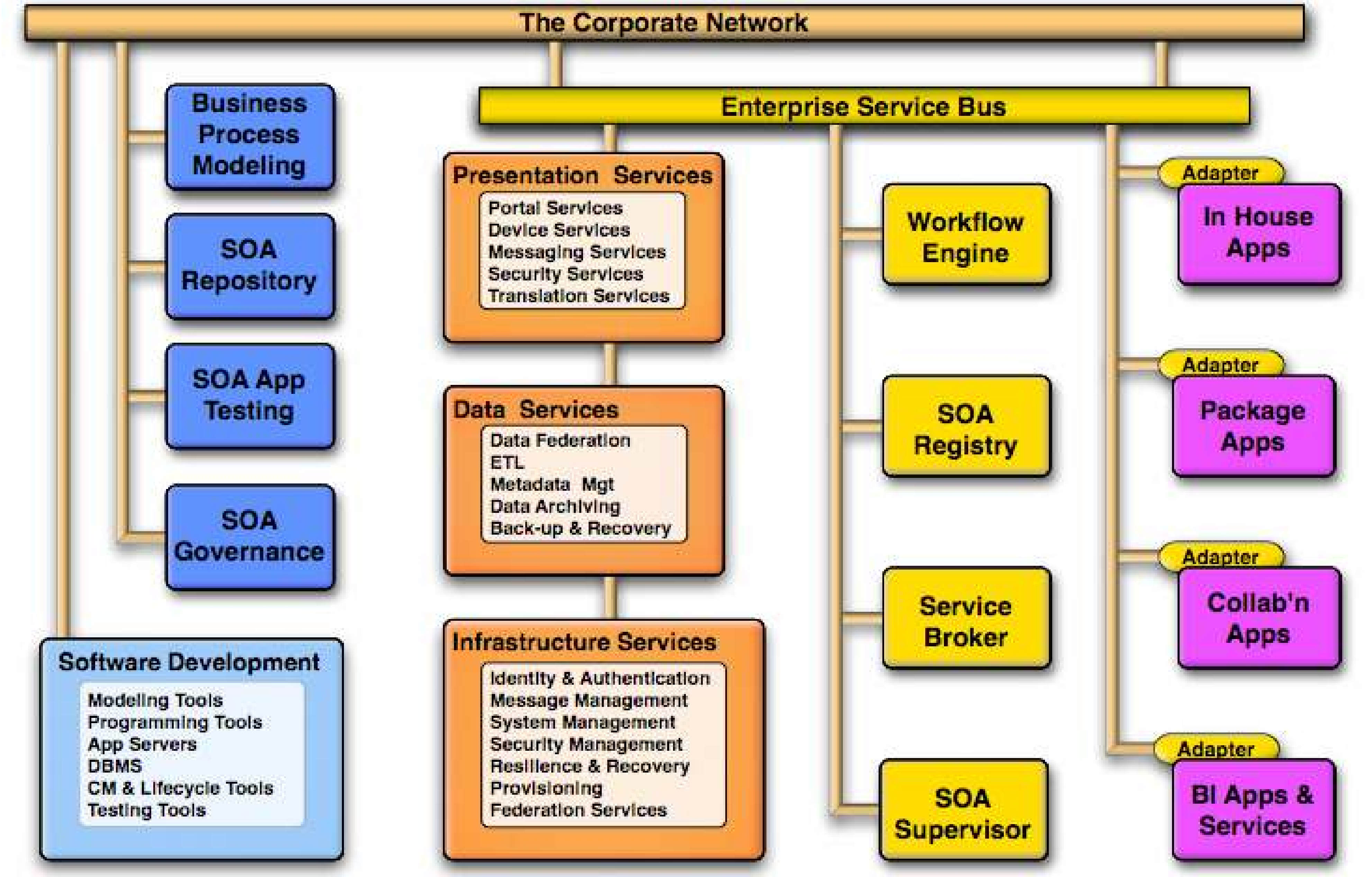
What Principles does SOA use to enable Agility?

*J. Hurwitz, SOA for Dummies, 2006

Service Orientation Principles

- **Loose Coupling**
  - Services maintain a relationship that minimizes dependencies and only requires that they retain an awareness of each other
- **Service Contract**
  - Services adhere to a communications agreement, as defined collectively by one or more service descriptions and related documents
- **Autonomy**
  - Services have control over the logic they encapsulate
- **Abstraction**
  - Beyond what is described in the service contract, services hide logic from the outside world
- **Reusability**
  - Logic is divided into services with the intention of promoting reuse
- **Composability**
  - Collections of services can be coordinated and assembled to form composite services
- **Statelessness**
  - Services minimize retaining information specific to an activity
- **Discoverability**
  - Services are designed to be outwardly descriptive so that they can be found and assessed via available discovery mechanisms
- **Governable**
  - Service Artifacts are visible and are able to be versioned, deprecated, and otherwise managed based on the proper level of authentication and authorization

An Overall Model for SOA



The Corporate Network

Business Process Modeling

SOA Repository

SOA App Testing

SOA Governance

Software Development

- Modeling Tools
- Programming Tools
- App Servers
- DBMS
- CM & Lifecycle Tools
- Testing Tools

Enterprise Service Bus

Presentation Services

- Portal Services
- Device Services
- Messaging Services
- Security Services
- Translation Services

Data Services

- Data Federation
- ETL
- Metadata Mgt
- Data Archiving
- Back-up & Recovery

Infrastructure Services

- Identity & Authentication
- Message Management
- System Management
- Security Management
- Resilience & Recovery
- Provisioning
- Federation Services

Workflow Engine

SOA Registry

Service Broker

SOA Supervisor

Adapter

In House Apps

Adapter

Package Apps

Adapter

Collab'n Apps

Adapter

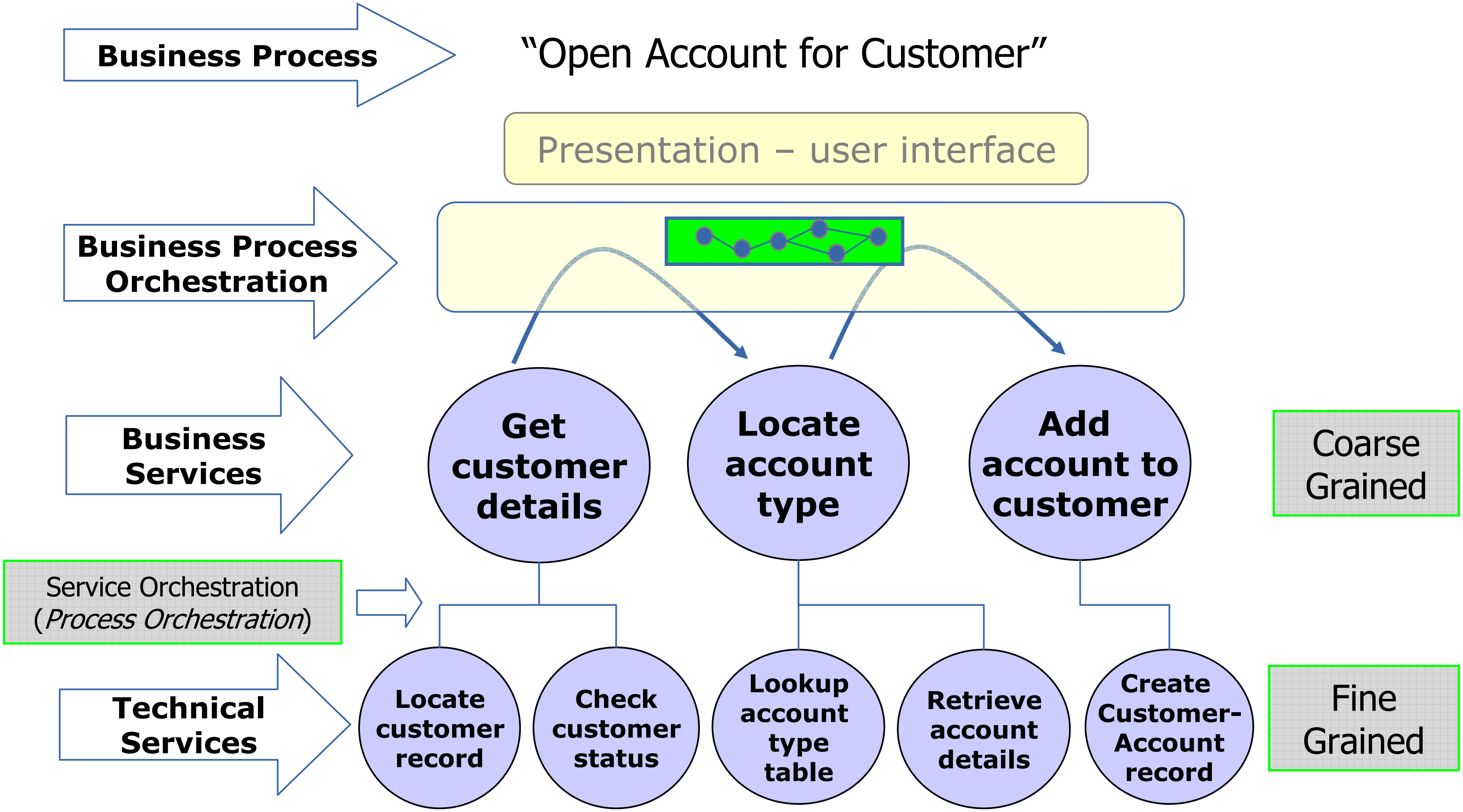
BI Apps & Services

SOA UNIVERSE

SOA Knowledge in Action

Source: SOA for Dummies/Hurwitz Associates

Business Processes & Services



Business Process

"Open Account for Customer"

Presentation – user interface

Business Process Orchestration

Business Services

Get customer details

Locate account type

Add account to customer

Coarse Grained

Service Orchestration (*Process Orchestration*)

Technical Services

Locate customer record

Check customer status

Lookup account type table

Retrieve account details

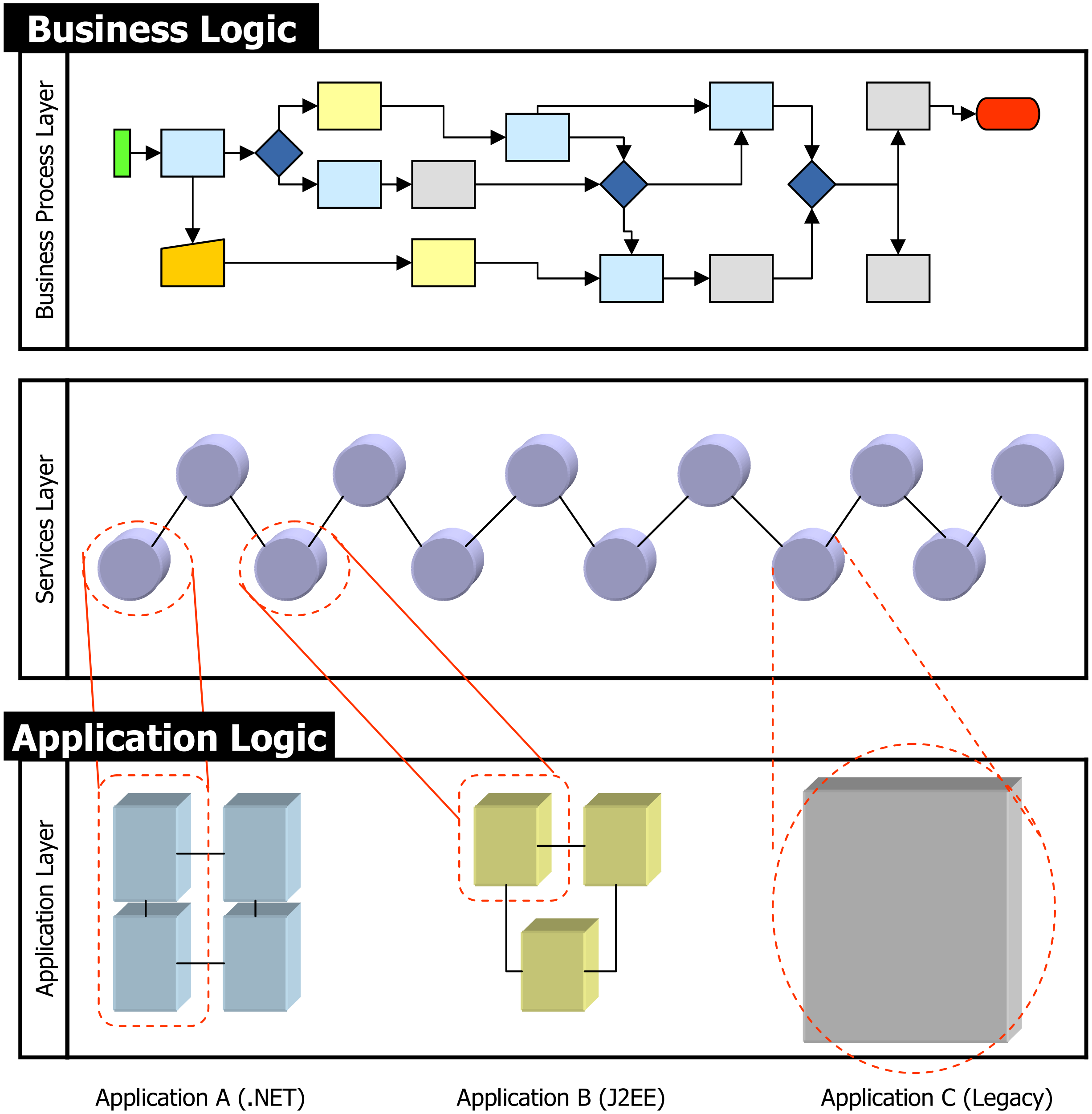
Create Customer-Account record

Fine Grained

Adapted from ANZ Banking Group Australia

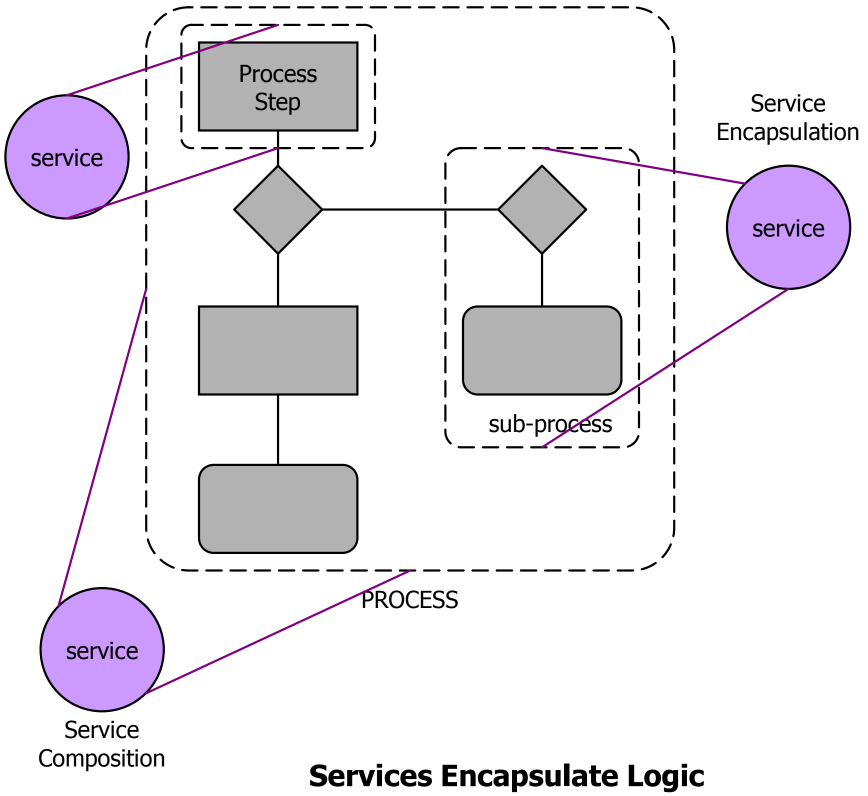
Service-Orientation & the Enterprise

- The collective logic (or processes) that defines and drives the enterprise is an evolving entity that changes in response to external & internal influences
- From an IT perspective, this enterprise logic can be divided into 2 important halves: business logic and application logic
- *Business Logic*: processes that express business requirements
- *Application Logic*: implementation of business logic organized into technology solutions
- **Services establish an abstraction layer wedged between traditional business & application layers.**
- Services are individually responsible for the *encapsulation* of specific application logic.



1 2 3

Fundamental SOA consists of Services, Descriptions & Messages



Services Encapsulate Logic

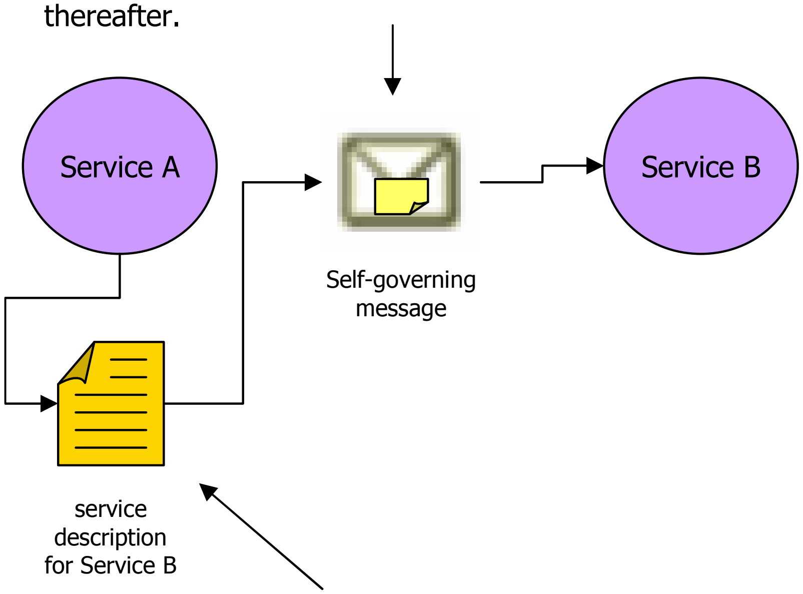
1

- In SOA, units of logic are known as **services**.
- To retain their **independence**, services encapsulate logic within a distinct context. This context can be specific to a business task or some other logical grouping.
- The concern addressed by a service can be large or small. Therefore, the size and scope of the logic represented by the service can vary.
- A **collective** is composed of several services.

Services Communicate Through Messaging

3

- Messages are "independent units of communication" which need to be outfitted with enough intelligence to self-govern their parts of processing logic.
- Similar to services, messages must be autonomous since after a service sends a message on its way, it loses control of what happens to the message thereafter.



Services Relate Through Service Descriptions

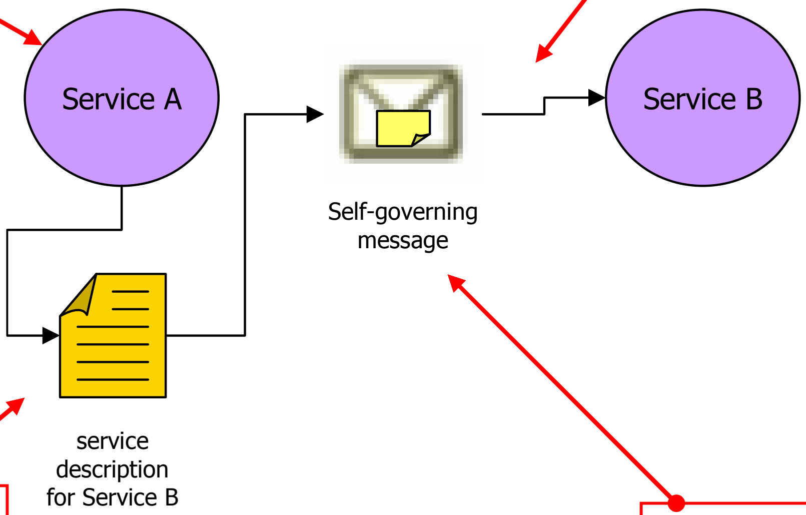
2

- In SOA, services are **aware** of each other through the use of **service descriptions**.
- A service description establishes the name of the service and the data expected and returned by the service.
- The manner is which services use service descriptions results in a relationship classified as **loosely coupled**.

Critical Design Issues Require (*Service-Oriented*) Principles

How should services be designed?

How should the relationship between services be defined?



How should service descriptions be designed?

How should messages be designed?

SOA Definitions: Service

A Service :

- Is a unit of enterprise functionality provided by a technology that can be accessed by other technologies
- Is implemented (provided) by a provider, which describes and publishes a well-defined contract for use (consumption) by one or more consumer(s)
- Can be made available via a catalog or service directory and can be accessed in a static or dynamic manner (depending upon the implementation)

Some Characteristics of Services:

- The services are self-contained and do not depend on the context or state of the consumer or consuming service
- The interface contract to a service should be platform-independent
- A Service's Quality needs to be proportional to the importance of its use

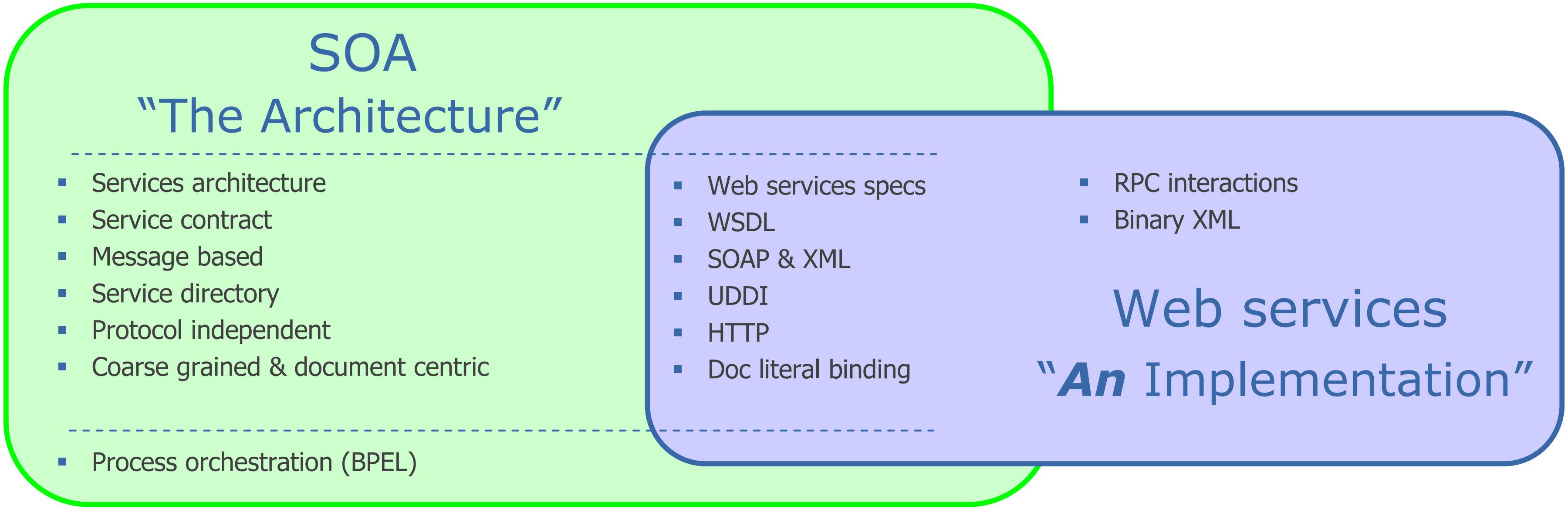
- Example : Credit Card Authorization (Service)

SOA Definitions : Business Process & Choreography

- A **Business Process** is an ordered (and usually long-running) sequence of activities, performed according to some well-defined business rules and with a specific business objective that produces a specific service or product for a consumer of that process.
- Examples are Hire New Employee, Process Insurance Claim, etc.
- The sequencing, selection and execution of business-aligned operations is collectively termed **Process Choreography**, and is maintained external to the services that participate in it.
- In typical SOA, choreographed services are invoked in response to business events.

SOA & Web Services

- SOA can be implemented without Web services, and Web services can be used for non-SOA (e.g. RPC) interactions. However, Web services delivers key standards for implementing SOA.
- The WS-* family scales to meet integration challenges intra-enterprise (enterprise application integration [EAI]) and inter-enterprise (business to business [B2B]).
- XML is an ideal candidate for loosely coupled inter-application data sharing. XML is not self-describing, but XML Schema can be be used to constrain message layout and content.



SOA

"The Architecture"

- Services architecture
- Service contract
- Message based
- Service directory
- Protocol independent
- Coarse grained & document centric

- Web services specs
- WSDL
- SOAP & XML
- UDDI
- HTTP
- Doc literal binding

- RPC interactions
- Binary XML

Web services

"*An* Implementation"

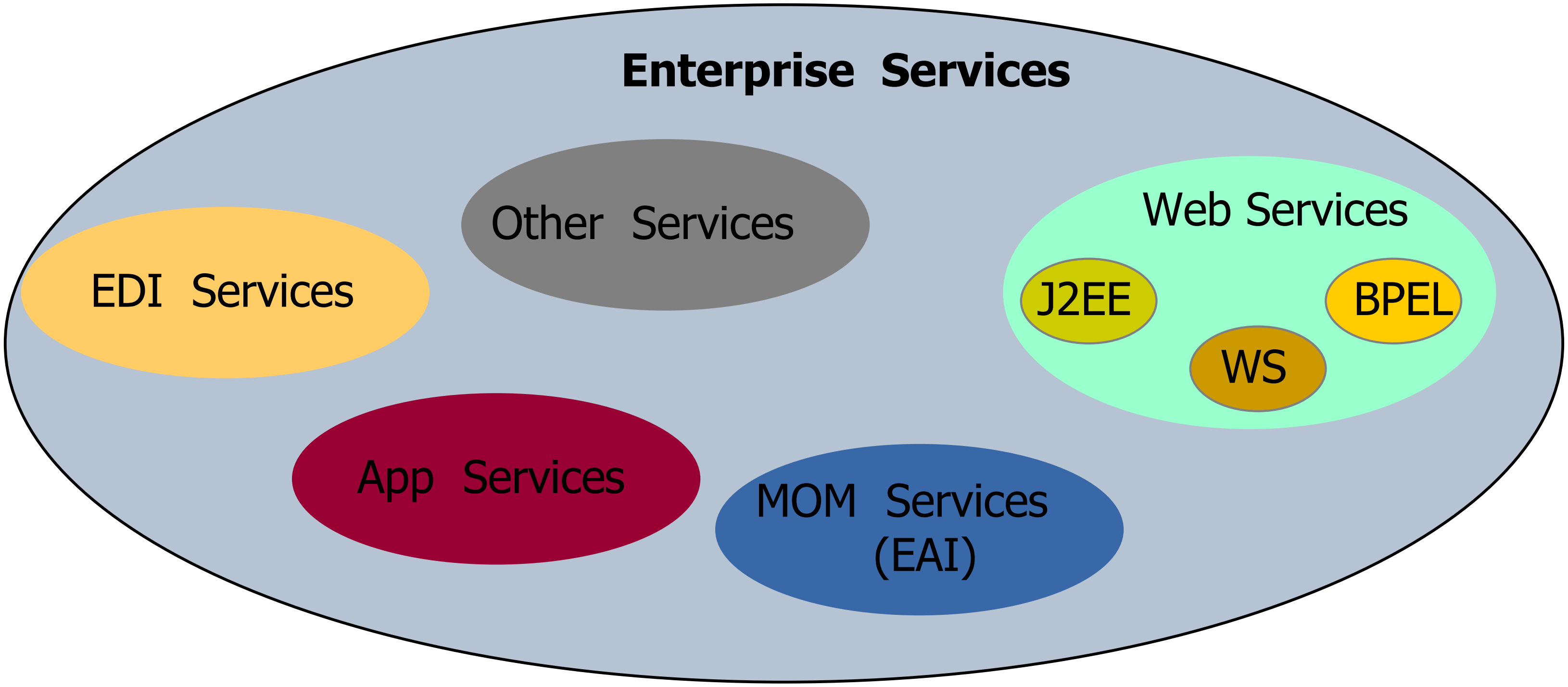
- Process orchestration (BPEL)

You don't have SOA until you build/buy services and compose them to implement business functionality.

Web Services is the stack of standard web technologies required at both consumer and provider ends to implement the pipe for shipping XML messages between them.

SOA Definitions: Enterprise Services & Web Services

- There are different kinds of services. A Web Service is one kind of implementation.
- In Web Services, the contract between provider and consumer, referred to as a **Service Interface**, is usually defined in some standardized and technology-independent notation such as **WSDL (Web Services Description Language)**,
- Integration with Enterprise Service is more complex since there is no standard integration interface
- SOA initiatives involve implementation of Enterprise Services which include Web Services, MOM Service etc., which need to work together.



Enterprise Services vs. Web Services

Web Service	Enterprise Service
All Web-Services are Enterprise Services	All Enterprise Services are not Web-Services
All Web-Services have a well defined service endpoint explaining request and response message structure	All Enterprise Services do not need to have an explicit message structure
All web-services have an open standard interface definition (WSDL)	All Enterprise Services do not have an open standard interface
All web-services are XML based	All Enterprise Services are not strictly XML based
All web-service endpoints are defined via WSDL	Enterprise Services endpoint can be defined by: API, MQ, File, FTP, TCP/IP,SOAP, JMS, and so on…
All Web-Service endpoints are implemented as software although some hardware devices could support software extensions.	Enterprise Service endpoints can be implemented by hardware or software

SOA and Standards

- A commonly cited requirement for classical SOA is that it be founded on open standards like XML, WSDL, etc. and conform to the WS-X compatibility and interoperability specifications.

- However, such standards are by no means a **mandatory requirement** for a generic SOA approach: in fact, Web Services are the only class of services that actually require XML schemas and WSDL bindings for their operation.

- The use of standards merely enables and facilitates flexible re-usable services, rather than being counted among its core attributes alongside those listed above.

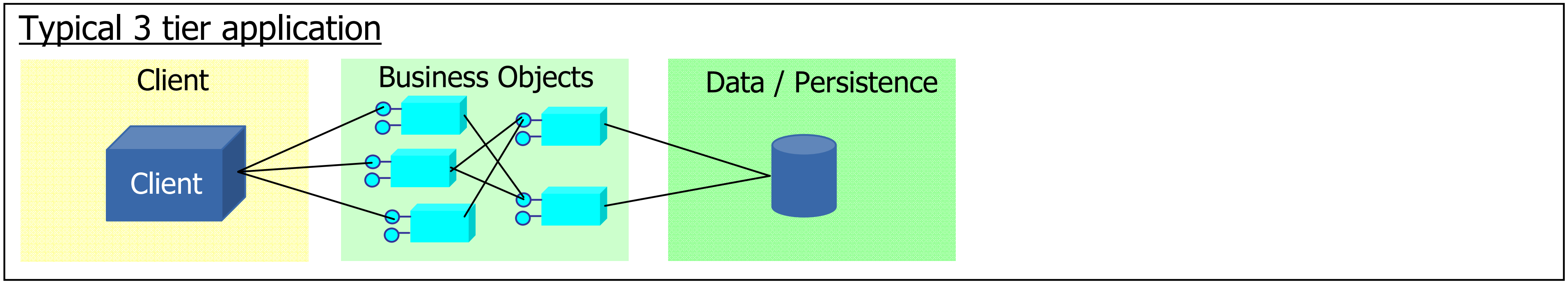
- While it is a **popular myth** that Web Services are synonymous with SOA, their advent merely helped to promote and make the latter a mainstream idea, and is just one of its many vehicles.

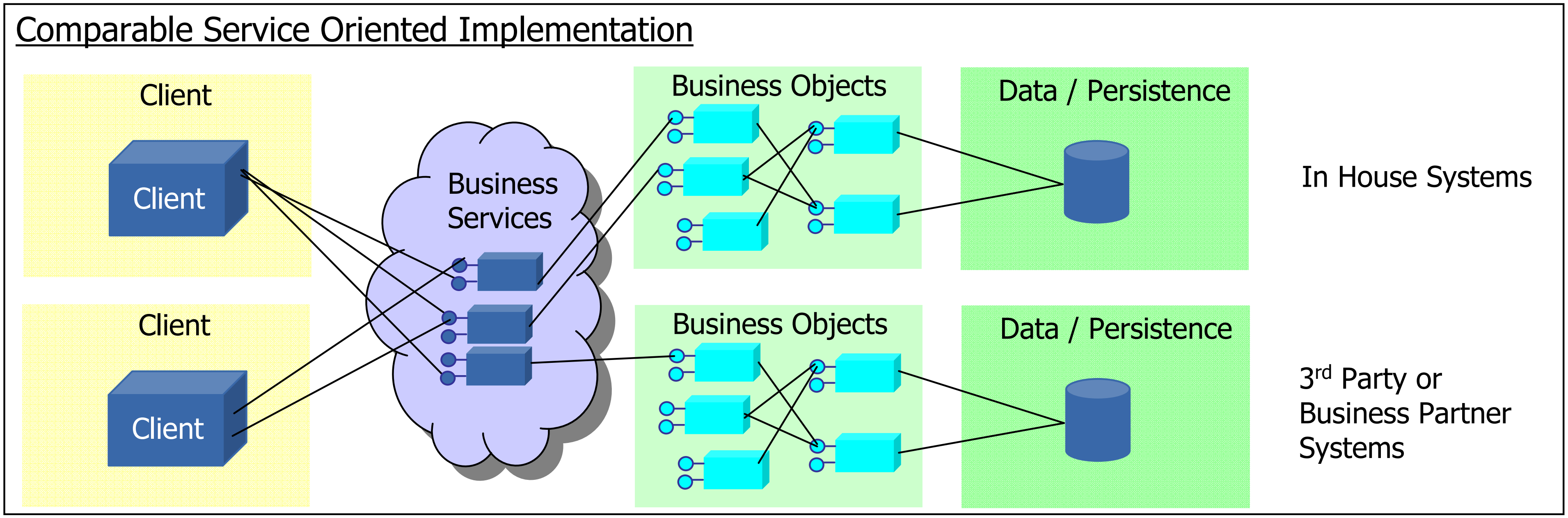
Key Definitions

- Services
  - Reusable components encapsulating business or functional capability
- Service Oriented Architecture (SOA)
  - An approach to using services in a technology approach to building business software
- Web Services
  - XML-based platform-independent standard to pass data and call processes within distributed systems
- Enterprise Services Architecture (ESA)
  - Architecture based on SOA and applied to the whole enterprise, both business and technology
- Business Process Management/Modeling (BPM)
  - Standards, techniques and tools focused on modeling processes in a way that allows for process definition in services and process enablement through orchestration
- Orchestration
  - Assembly and execution of services following a business process
- Choreography
  - Multi-party collaboration using services
- Composite Services
  - A new service built out of disparate services

How Does SOA Work?

- In a Service Oriented Architecture
  - SOAs build on previous 3-tier architectures to become more flexible, less redundant
  - The services then provide black-box functionality for business processes with location transparency
  - Location transparency provides the window of opportunity to incrementally sunset legacy applications and/or to access 3rd party services





Source: Adapted from Microsoft

Why Consider a Service Oriented Architecture?

Focus on the Business

- Define business rationale, not technical features
- Find a pragmatic balance between technical rigor and time-to-market
- Value ongoing flexibility and agility over a one-time efficiency gain

Increase ROI

- With a little planning, you can get immediate return for each service you build via the "Network Effect"
- And you can get increasing return as your architecture – and that of your customers, suppliers, and partners – evolves to SOA

Achieve Reuse

- SOA is about reuse of existing assets: Legacy, Client Server, and Web
- You can "wrap" existing applications, re-using existing functionality of legacy systems to increase their reach and longevity
- And build new services on multiple supported platforms

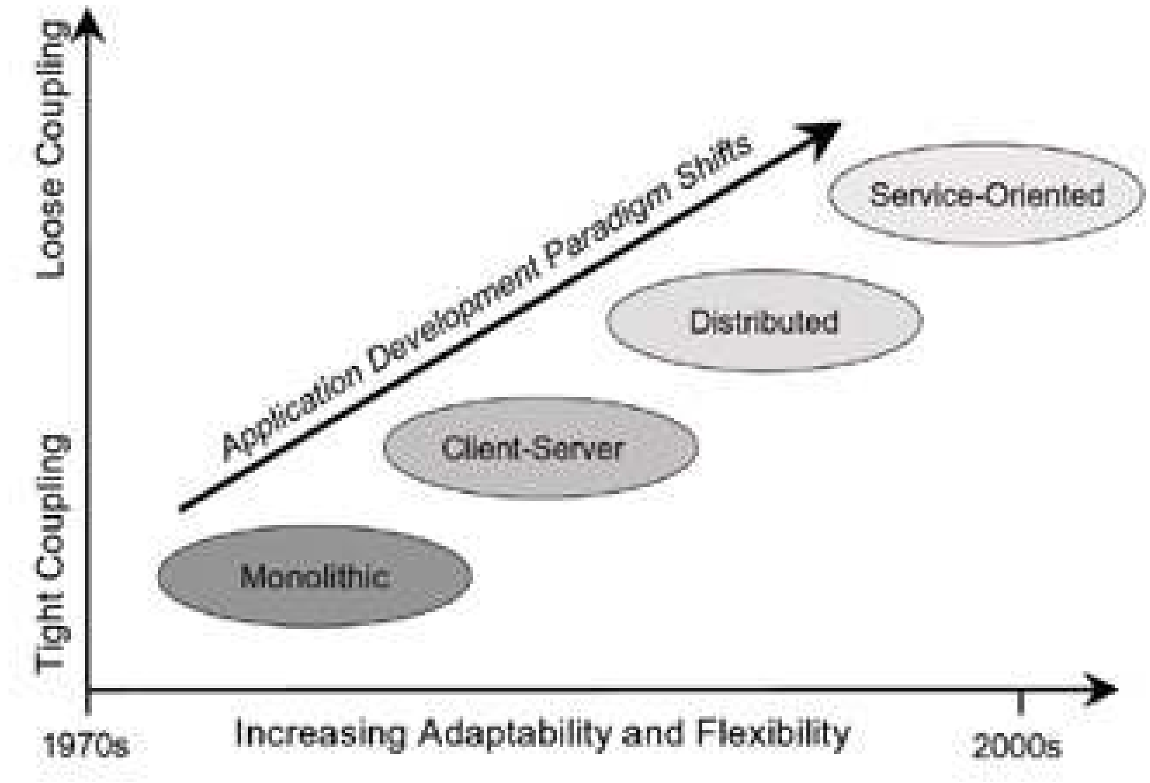
Future-Proof Your Enterprise

- Invest in a diversified portfolio of applications, not a single packaged application or a technology platform
- Applications are less fragile, more adaptive to rapidly changing business requirements
- Facilitate standards based integration with trusted business partners (B2B)
- Ease integration needs necessitated by M&A activity

Ensure Flexibility

- Complexity is encapsulated
- Code is mobile
- Enhancements and changes can be added incrementally without a negative ripple effect across the application infrastructure

SOA: A Natural Evolution of Application Development



Service Orientated Platforms Align Business Processes with Course Grained Services

Use a Collection of Components where replacing one or more components will not affect the applications that use the service

- ✓ In both Monolithic and Client/Server Systems, Making Changes Created Significant Ripple Effects Across the Enterprise Application Suite
- ✓Service Oriented Applications Are more Flexible and More Easily Reconfigured. Much More Like "Changing the Tires While Traveling Down The Road at 50MPH"
- ✓Service Oriented Applications Are Implemented as Discrete Business Services and Allow More Flexibility in Mixing Both Hardware and Software Platforms

The SOA Paradigm Shift

Distributed Architecture		Service Oriented Architecture
Functionally Oriented	→	Process Oriented
Designed to Last	→	Designed to Change
Long Development Cycle	→	Interactive & Iterative Development
Cost Centered	→	Business Centered
Application Block	→	Service Orchestrations
Tight Coupling	→	Agile & Adapted
Homogenous Technology	→	Heterogeneous Technology
Object Oriented	→	Message Oriented
Known Implementation	→	Abstraction

What is SOA? Discussion Questions

- How does these Models Fit with your Experiences?
- Does your Organization have a deep Understanding of SOA Concepts?
- If not, what are the Key Obstacles to Understanding?

SOA UNIVERSE

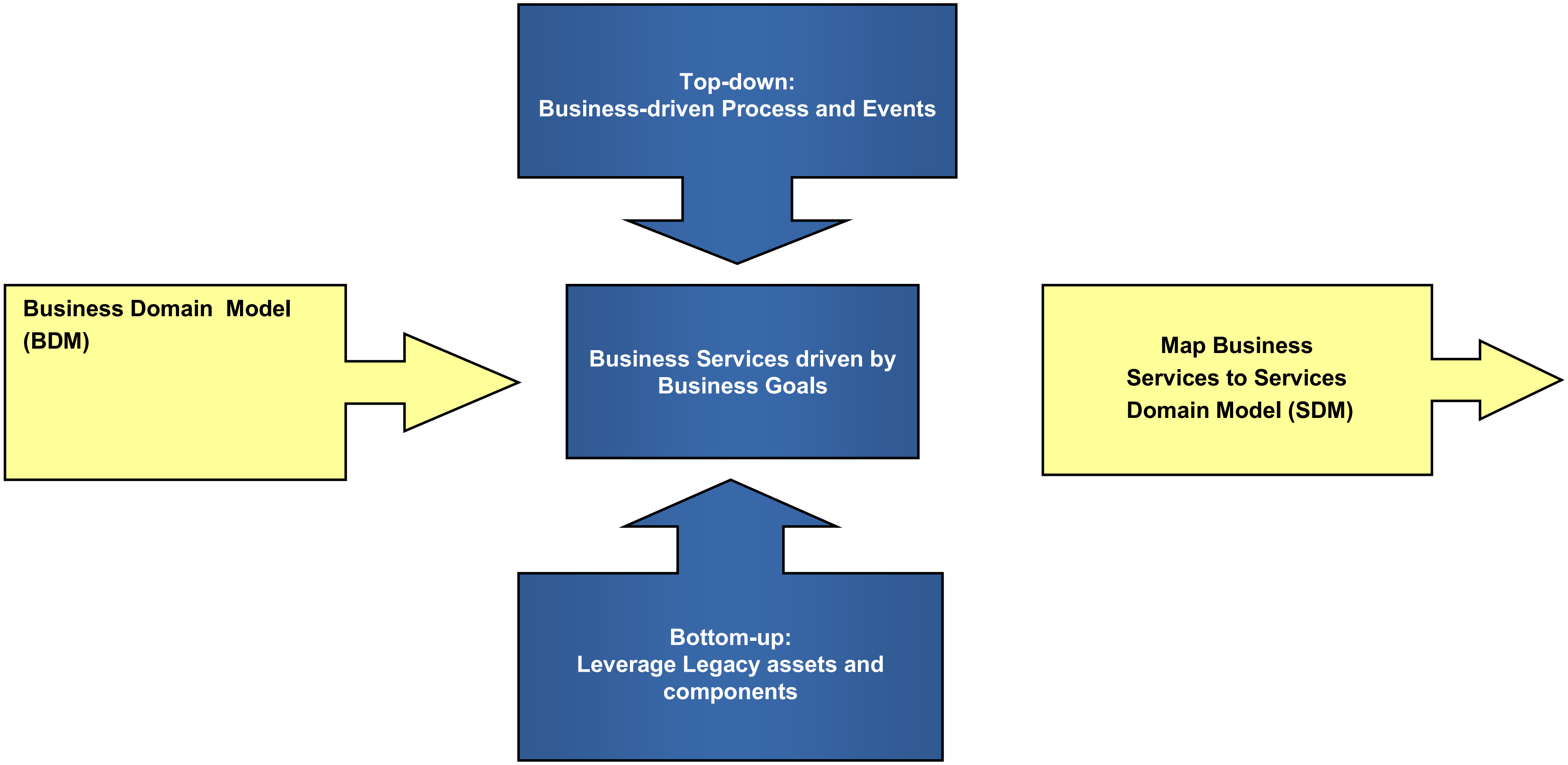
SOA Knowledge in Action

HURWITZ & ASSOCIATES

liquidhub

Section II: Business Architecture and SOA

SOA Mandates Understanding the Business



Top-down:
Business-driven Process and Events

Business Domain Model (BDM)

Business Services driven by Business Goals

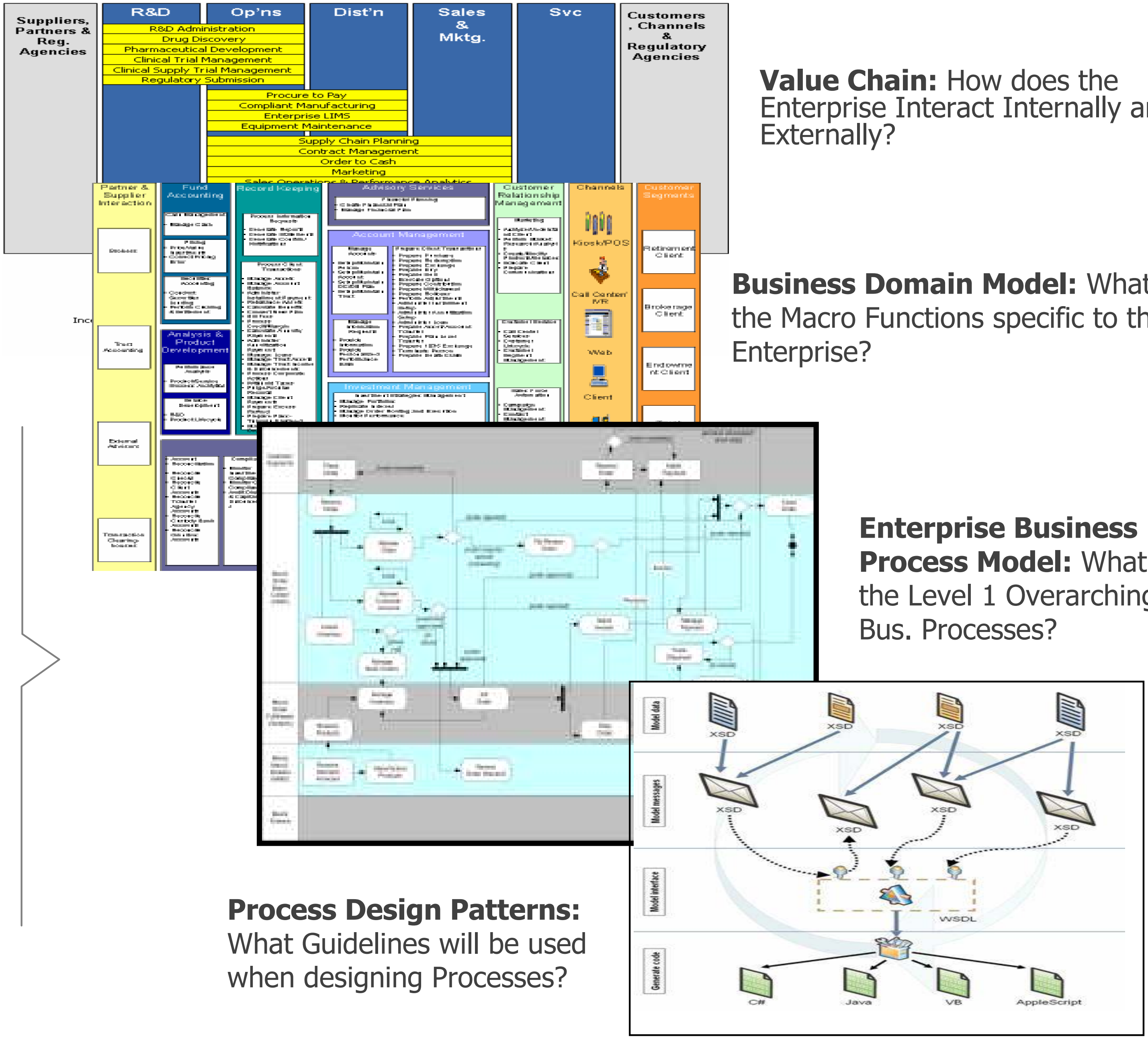
Map Business Services to Services Domain Model (SDM)

Bottom-up:
Leverage Legacy assets and components

High Value and Alignment can be achieved by decomposing Business Processes and Formulating an Actionable Strategy and Roadmap

Enterprise Business Services: From Value Chain to Services

Business Arch. Model



Value Chain: How does the Enterprise Interact Internally and Externally?

Business Domain Model: What are the Macro Functions specific to the Enterprise?

Enterprise Business Process Model: What are the Level 1 Overarching Bus. Processes?

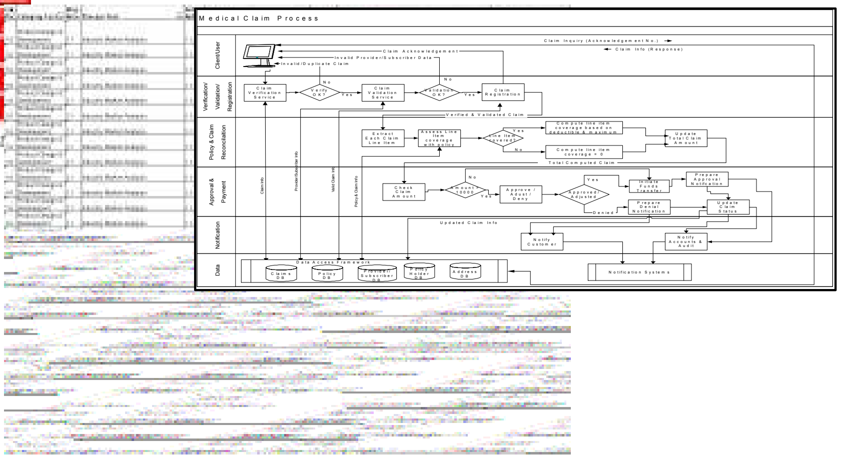
Services Domain Model

Process Design Patterns: What Guidelines will be used when designing Processes?

Enterprise Business Services: Roadmap

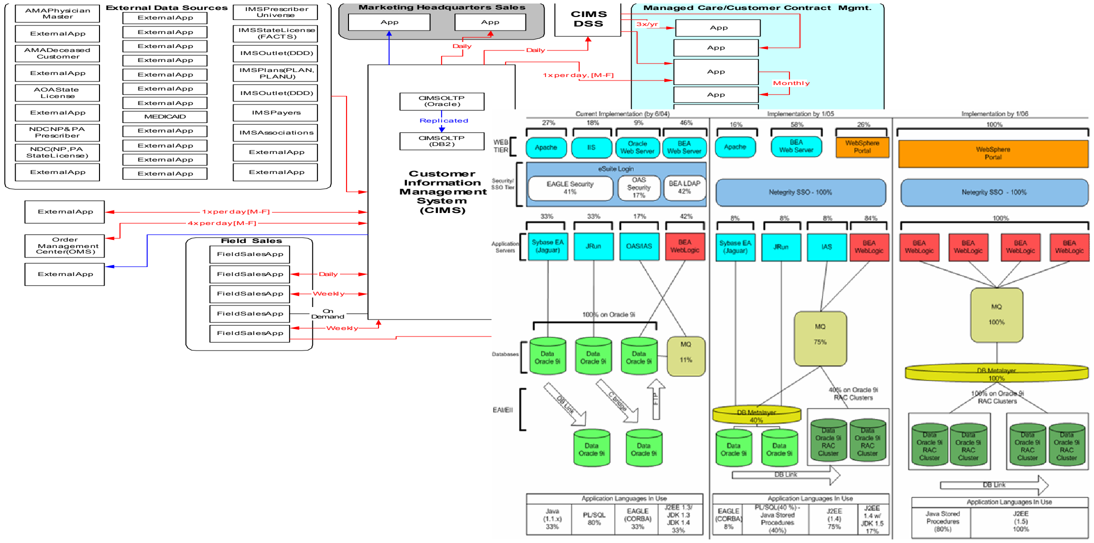
Services Design

Application Mapping: Given our Service Domain Model, what Application Overlaps and Gaps Exist?


Medical Claim Process
Client/User
Verification/ Validation/ Registration
Policy & Claim Reconciliation
Approval & Payment
Notification
Data

Utility Services Identification, Level 2/3 Processes, Data Specification: What is the detailed list of Specific Business and Utility Services, and how does Enterprise Data fit in?

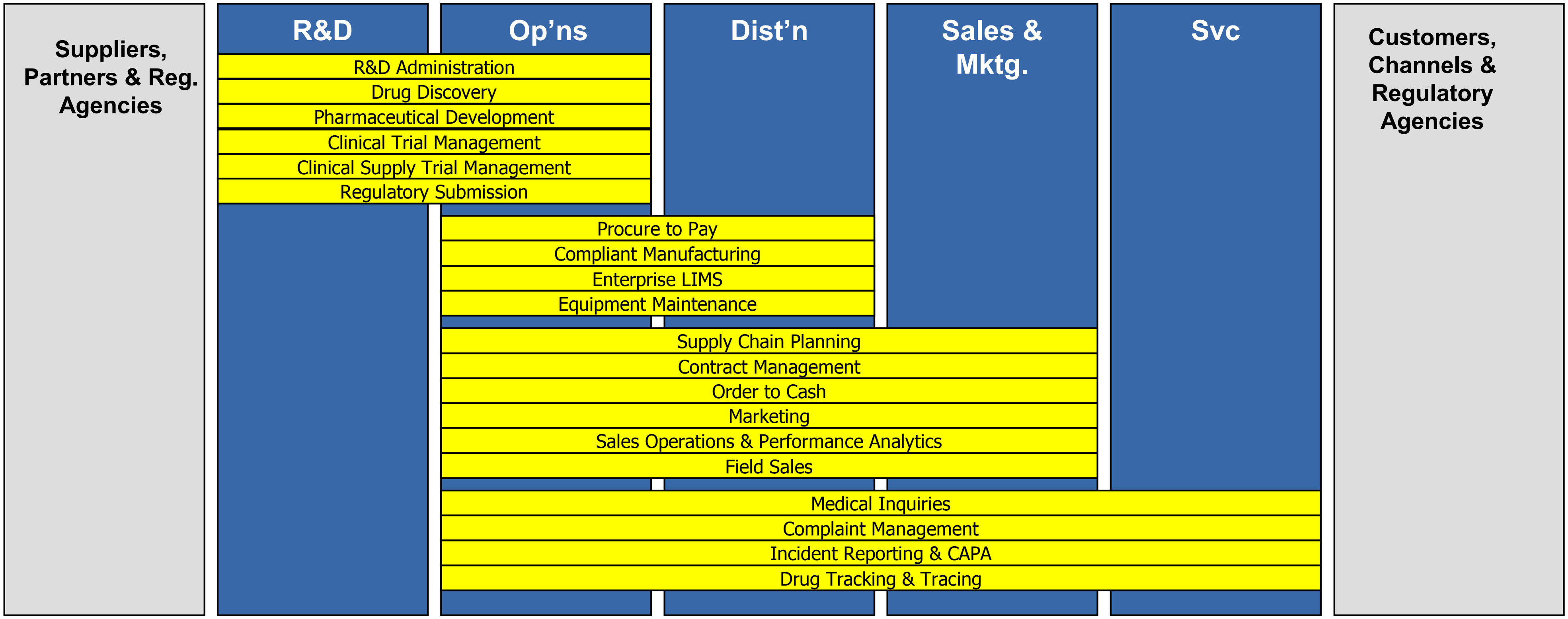
Business Process Model



Application Integration Design, Sol'n Architecture: What are the details of the Services to be built and deployed?

Business Architecture

The **Business Value Chain** provides a simple way to capture and organize all of the “macro” functions of the Organization and becomes the input to the next mapping step, the full **Business Domain Model**



Suppliers, Partners & Reg. Agencies

R&D

Op’ns

Dist’n

Sales & Mktg.

Svc

Customers, Channels & Regulatory Agencies

R&D Administration
Drug Discovery
Pharmaceutical Development
Clinical Trial Management
Clinical Supply Trial Management
Regulatory Submission

Procure to Pay
Compliant Manufacturing
Enterprise LIMS
Equipment Maintenance

Supply Chain Planning
Contract Management
Order to Cash
Marketing
Sales Operations & Performance Analytics
Field Sales

Medical Inquiries
Complaint Management
Incident Reporting & CAPA
Drug Tracking & Tracing

FINANCIALS					
General Ledger	Fixed Asset Acct	Tax Acct	Cost Center Acct	Product Cost Acct	EP&P
A/R	Cash Journal Acct	Financial Statements	Project Acct	Transfer Pricing	Collections Mgmt
A/P	Inventory Acct	Profit Center Acct	Investment Mgmt	Credit Mgmt	Risk Mgmt

CORPORATE SERVICES
Incentive & Commission Management
Real Estate Management
Indirect Procurement
Travel Management

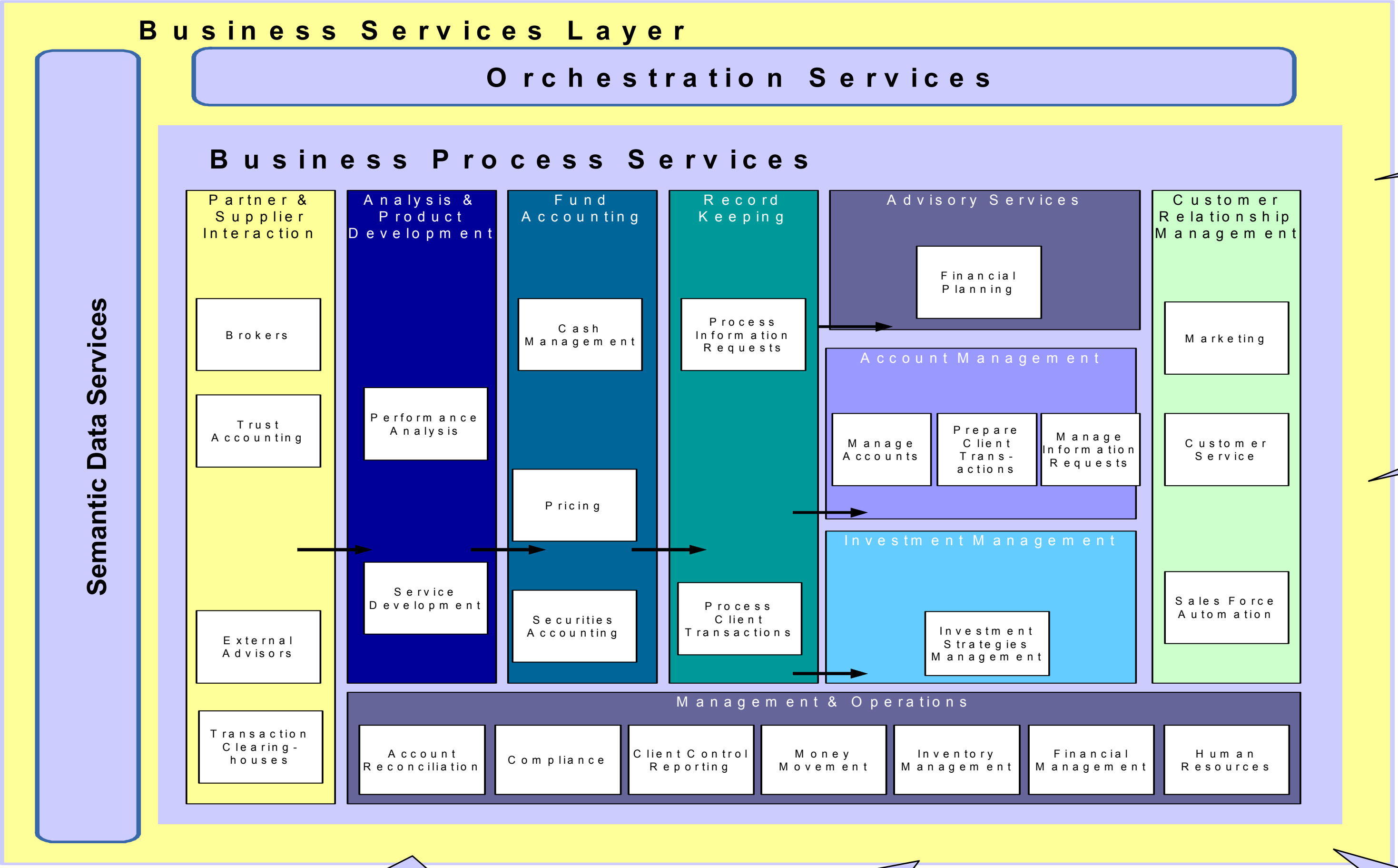
HUMAN CAPITAL MANAGEMENT	
Recruiting	Payroll & Legal Reporting
Career Management	Resource Management
Enterprise Learning	Benefits Management
Performance Management	Time & Attendance

CORPORATE GOVERNANCE
Audit Information Systems
Mgmt of Internal Controls
Business Risk Mgmt
Transparency for Basel II

Sample Pharmaceutical Business Value Chain (Source: LiquidHub)

Business Domain Modeling

Business Domain Models (BDM) are used to show how the various ”macro” functions captured in the Value Chain Model can be further decomposed and grouped. Critical questions further drive the definition of the **Service Domain Model**.



Business Services Layer

Semantic Data Services

Orchestration Services

Business Process Services

Partner & Supplier Interaction
Brokers
Trust Accounting
External Advisors
Transaction Clearing-houses

Analysis & Product Development
Performance Analysis
Service Development

Fund Accounting
Cash Management
Pricing
Securities Accounting

Record Keeping
Process Information Requests
Process Client Transactions

Advisory Services
Financial Planning

Account Management
Manage Accounts
Prepare Client Trans-actions
Manage Information Requests

Investment Management
Investment Strategies Management

Customer Relationship Management
Marketing
Customer Service
Sales Force Automation

Management & Operations
Account Reconciliation
Compliance
Client Control Reporting
Money Movement
Inventory Management
Financial Management
Human Resources

Where are the core and non-core activities/ processes? What are the current business priorities?

How does my current application portfolio map to the business architecture? Where are my application redundancies?

How should teams be organized in the future to support the process?

What are the priorities for Reusable Application Services?

What are the gaps in our Solution portfolio? Where is the opportunity to improve and automate processes?

Which processes are internal vs. external via partners, suppliers etc?

Sample BDM (Source: LiquidHub)

Service Domain Model

Helps identify and prioritize business services for possible implementation (initially at a high level); this is an Iterative process that helps to focus on high-value services as ranked by:

- business value and impact
- reuse and high consumption
- Feasibility
- technical viability

From these prioritized business services, a set of services will be selected for implementation based on project needs, business imperatives and other organizational requirements

Identifying Business Services

- The services identification process should be conducted in top-down fashion initially by a joint Business and Enterprise Architecture team.
- Current technical environment is not important at this stage!
- Challenge is to identify the business services in the enterprise within some initial scope to set priority and to model and design the services

5 Viable Methods to Identify Candidate Business Services:

- Business Process Analysis: Are there existing process diagrams and doc’n?
- Core Entity Analysis: What are the major “Nouns” (Actors/Data) of the business?
- Budgeted Initiatives Analysis: What in-process initiatives are good fits to leverage Service work?
- Preexisting Services Analysis: Are there existing Services that can be “harvested”?
- Existing Business Applications Analysis: Which Applications fit the Services Model now?

Services Design Best Practices

The Services Design Phase is where the low-level details of the actual Business Services are collected and laid out for development (and/or "assembly"). Pragmatically, this phases matches up existing applications, services, and projects and exposes the gaps that require development of new services and/or modification of existing ones.

Industry best-practices highlight 2 basic ways to undertake Services Design:

- Top-Down Based Modeling and Design
  - Behavior "Use Case" First
  - Interface (Document) First
- Bottom-Up Based Modeling and Design
  - *Audit*
    - Service Realization
    - Transformation Realization
    - Service Derivation
  - *Gap Analysis*
    - Service Definition
    - Cost vs. Benefit

Services Design: Top-Down "Use Case First" Modeling Approach

- As implied by its name, it involves modeling the process behavior first
- Starting from highest level, we begin modeling business rules as they are encountered in a Use-Case
- While success and failure Use-Cases are documented separately, when creating a process model, we often combine these as separate paths within the same process model
- As such, although there may be many use-cases, there is no one-to-one correlation between a use-case and process model. A single process model can describe many use-cases
- This mode of modeling is best suited for Senior Analyst or higher level of audience
- Once the behavior is defined, we focus on the data needed to support the behavior and derive the process interfaces from it
- While this approach is very good at deriving some details of services interfaces, it often leaves the interface details for modeling at a later time

Advantages	**Disadvantages**
Interfaces are more likely to be decoupled from specific business context where they are used	Invalid data granularity – The process model may not have been designed to handle the level of granularity that was modeled for an interface
	Irrelevant data requirements – Data model of an interface may require pieces of information that the Process Model does not have or need
	Complexity Of Interfaces – Resulting interfaces tend to be more complex

Typically, Top-Down Approaches are utilized in Strategic Planning Exercises..

Services Design: Bottom-Up “Audit” Approach

- Bottom-Up Approach almost always starts by auditing of current enterprise assets
- The auditing activity has three distinct aspects:
  - Service Realization
  - Service Derivation
  - Transformation Realization
- The Auditing of Assets leads to Gap Analysis
- The Gap analysis activity has following distinct aspects:
  - Service Definition
  - Cost vs. Benefit analysis
- Bottom-Up Audit Approaches are also employed to build hands-on experience and education of SOA principles within development teams

..While Bottom-Up Approaches are usually taken in Project or Point-Solution-based situations.

SOA Business Architecture: Discussion Questions

- How does this Approach Fit with your Experiences?
- What Visibility do you have into your Business Value Chain and Associated Service Mapping?
- How do you currently go about Setting Priority and Granularity of Service Design?
- Is your Services Design Approach “Top-Down”, or “Bottom-Up”?

SOA UNIVERSE

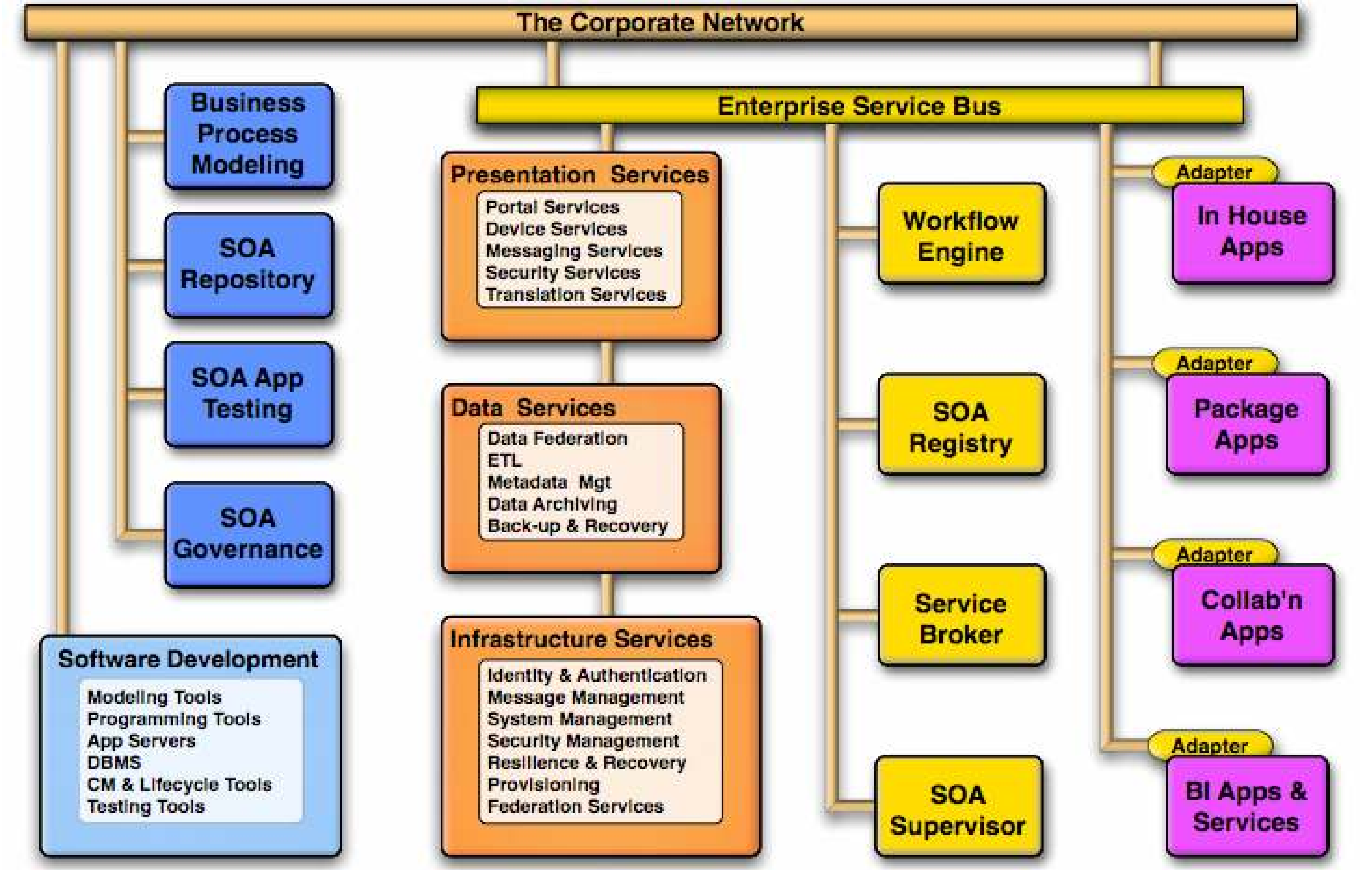
SOA Knowledge in Action

HURWITZ & ASSOCIATES

liquidhub

Section III: Technical Architecture of SOA

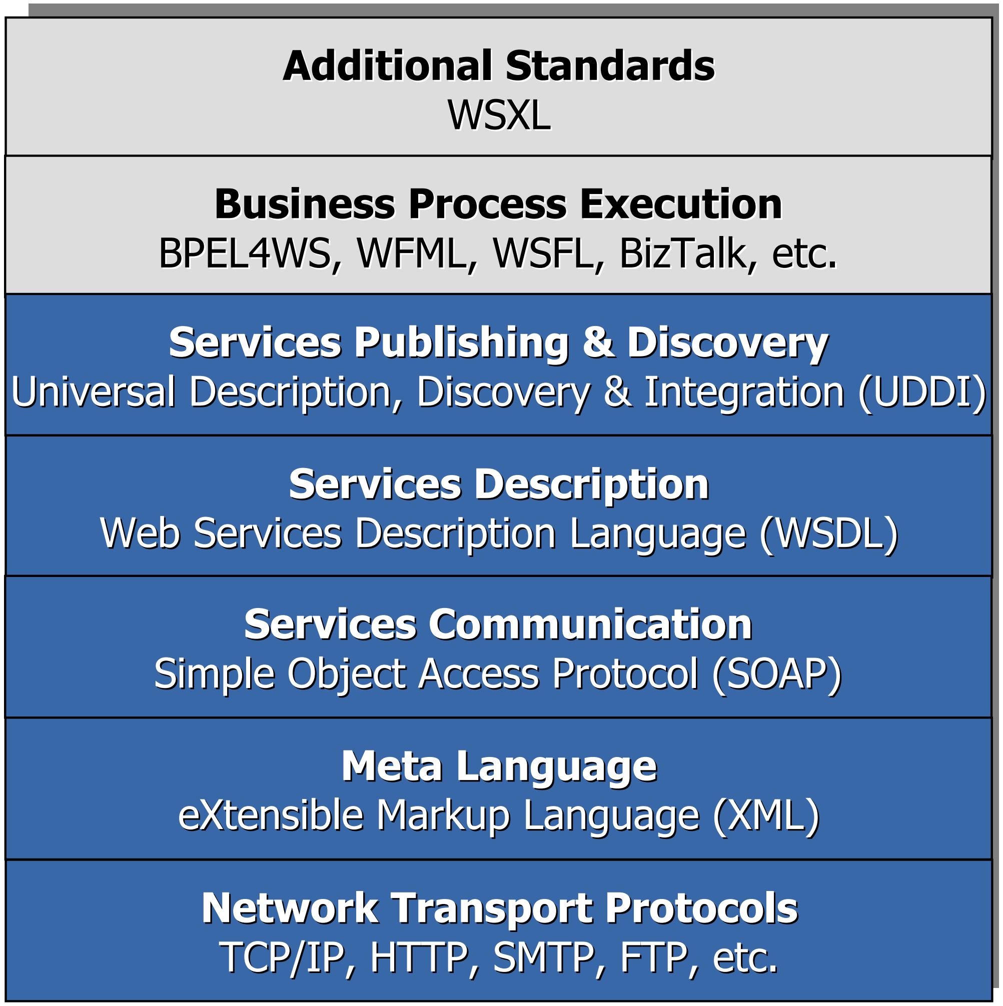
An Overall Model for SOA



Source: SOA for Dummies/Hurwitz Associates

Service Oriented Integration using Web Services

- The most popular approach due to use of open standards
- Based on ubiquitous Industry Standard Protocols with universal support
- Leverage the internet for low cost communications
- Deliver platform and technology independence
- Loosely Coupled
  - Migration from direct calls to architectural service reduces dependency on specific applications and packages
  - Supporting multiple application connection and information sharing scenarios
- Fosters re-use through publication of interfaces
  - Services are self describing
  - Reduces the time for developers to properly understand the interface
  - Richer specification of the service can be accessed programmatically
  - Reduces the impact of change
  - Provides dynamic service consumption



Additional Standards
WSXL

Business Process Execution
BPEL4WS, WFML, WSFL, BizTalk, etc.

Services Publishing & Discovery
Universal Description, Discovery & Integration (UDDI)

Services Description
Web Services Description Language (WSDL)

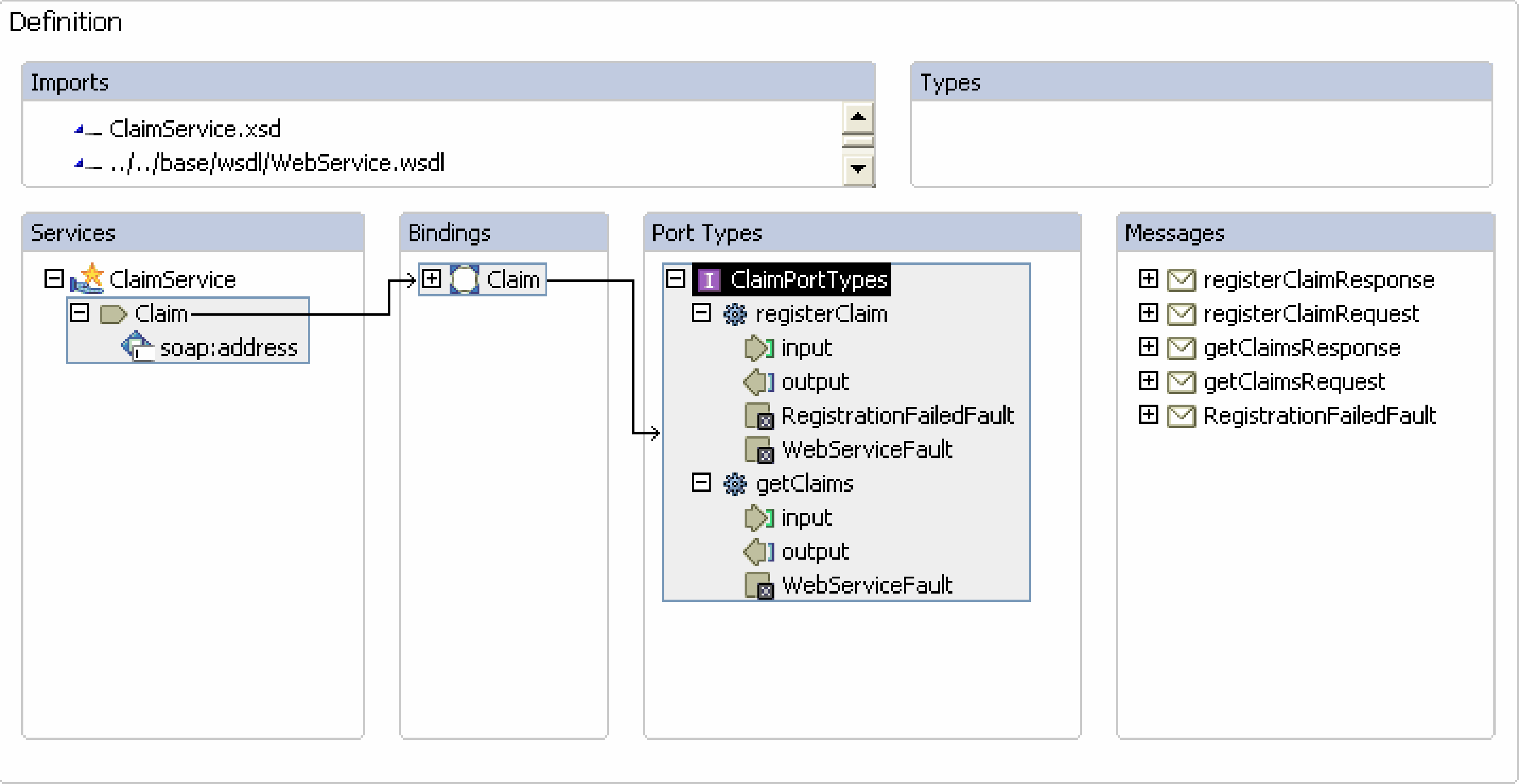
Services Communication
Simple Object Access Protocol (SOAP)

Meta Language
eXtensible Markup Language (XML)

Network Transport Protocols
TCP/IP, HTTP, SMTP, FTP, etc.

Source: CBDI

Web Services: *Service Interface Definition (WSDL)*



Definition

Imports

- ClaimService.xsd
- ../../base/wsdl/WebService.wsdl

Types

Services

- ClaimService
  - Claim
    - soap:address

Bindings

- Claim

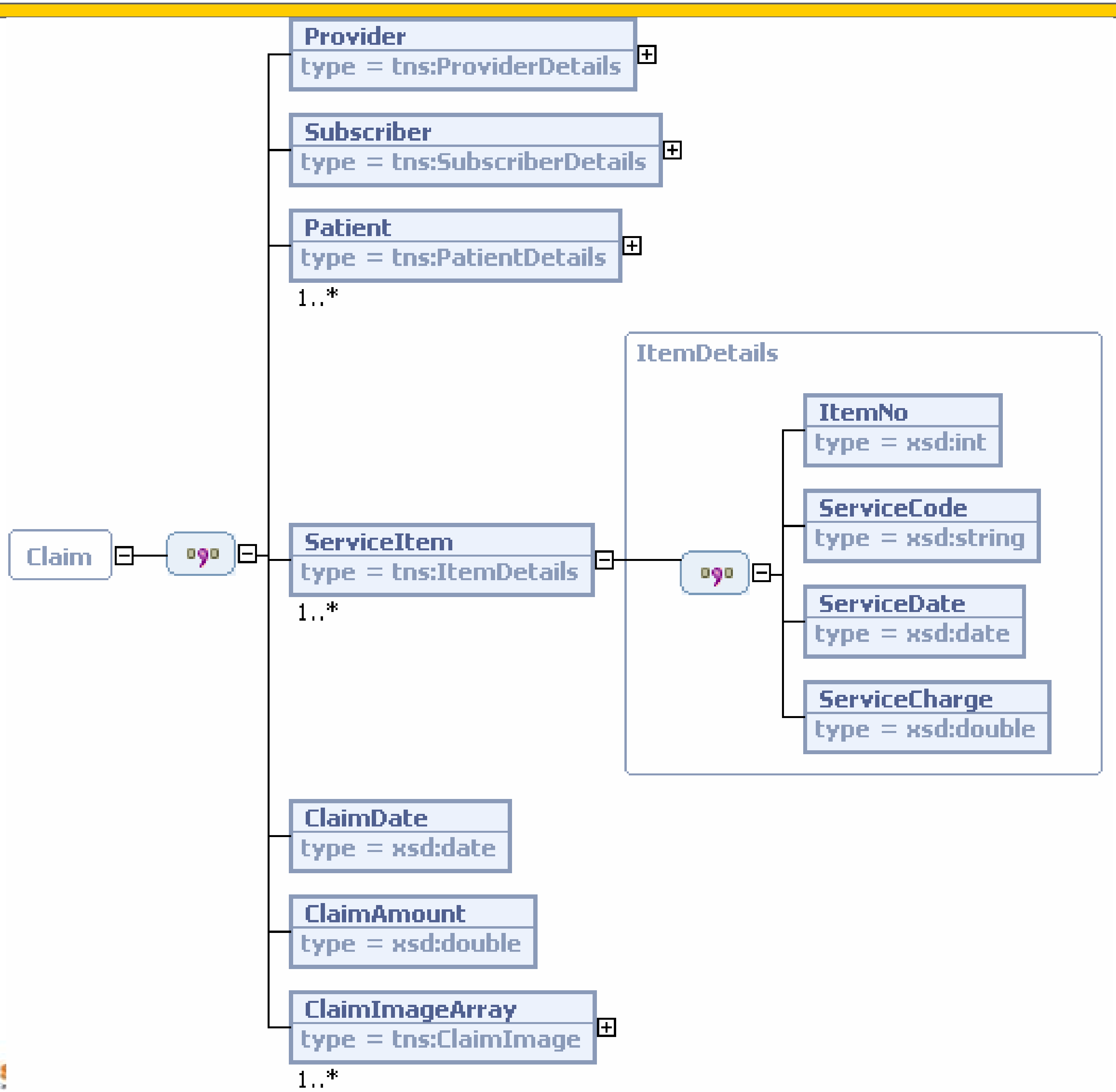
Port Types

- ClaimPortTypes
  - registerClaim
    - input
    - output
    - RegistrationFailedFault
    - WebServiceFault
  - getClaims
    - input
    - output
    - WebServiceFault

Messages

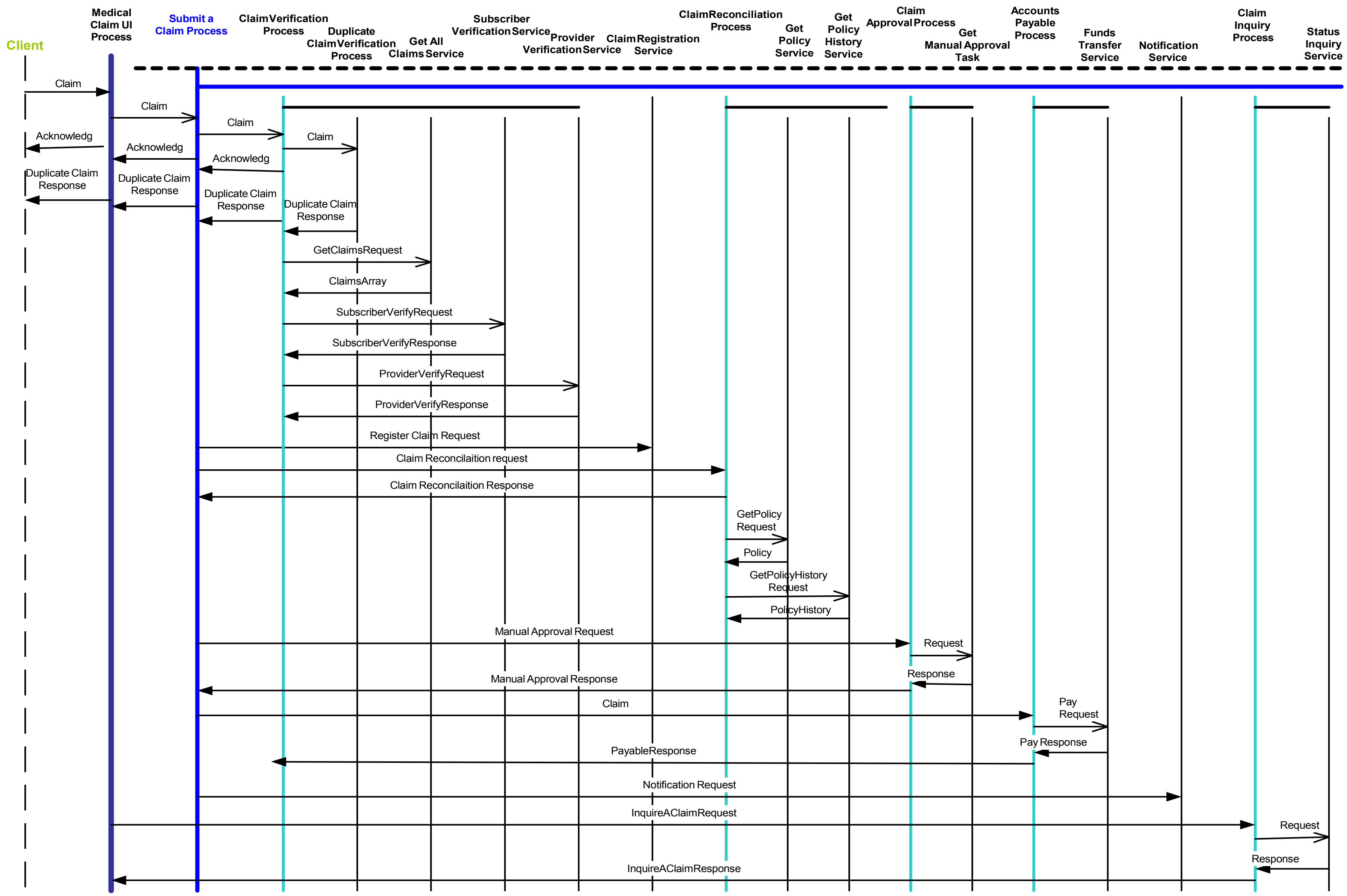
- registerClaimResponse
- registerClaimRequest
- getClaimsResponse
- getClaimsRequest
- RegistrationFailedFault

Web Services: *Schema Definition* (XSD)



Web Services: *Service Interaction Diagram*

Medical Claim Process - Process and Service Interaction Diagram



Client

Medical Claim UI Process

Submit a Claim Process

ClaimVerification Process

Duplicate ClaimVerification Process

Get All Claims Service

Subscriber Verification Service

Provider Verification Service

ClaimRegistration Service

ClaimReconciliation Process

Get Policy Service

Get Policy History Service

Claim Approval Process

Get Manual Approval Task

Accounts Payable Process

Funds Transfer Service

Notification Service

Claim Inquiry Process

Status Inquiry Service

Claim

Claim

Claim

Claim

Acknowledg

Acknowledg

Acknowledg

Duplicate Claim Response

Duplicate Claim Response

Duplicate Claim Response

Duplicate Claim Response

GetClaimsRequest

ClaimsArray

SubscriberVerifyRequest

SubscriberVerifyResponse

ProviderVerifyRequest

ProviderVerifyResponse

Register Claim Request

Claim Reconcilaition request

Claim Reconcilaition Response

GetPolicy Request

Policy

GetPolicyHistory Request

PolicyHistory

Manual Approval Request

Request

Response

Manual Approval Response

Claim

Pay Request

Pay Response

PayableResponse

Notification Request

InquireAClaimRequest

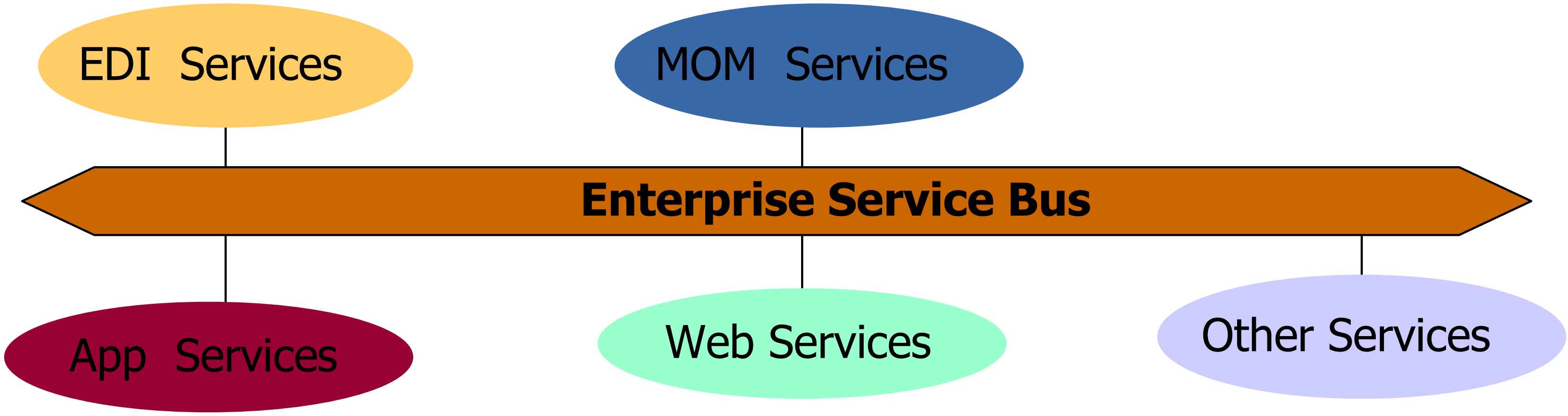
Request

Response

InquireAClaimResponse

Enterprise Services Bus (ESB)

- For a Service to be truly Enterprise Service, it must be accessible from all other Enterprise Services – All services must be able to talk to each other.
- An Enterprise Service Bus (ESB) is needed:
  - To integrate different kinds of Enterprise services
  - To leverage existing enterprise software assets
  - To facilitate a bridge between different types of communication, interface, and service end-points



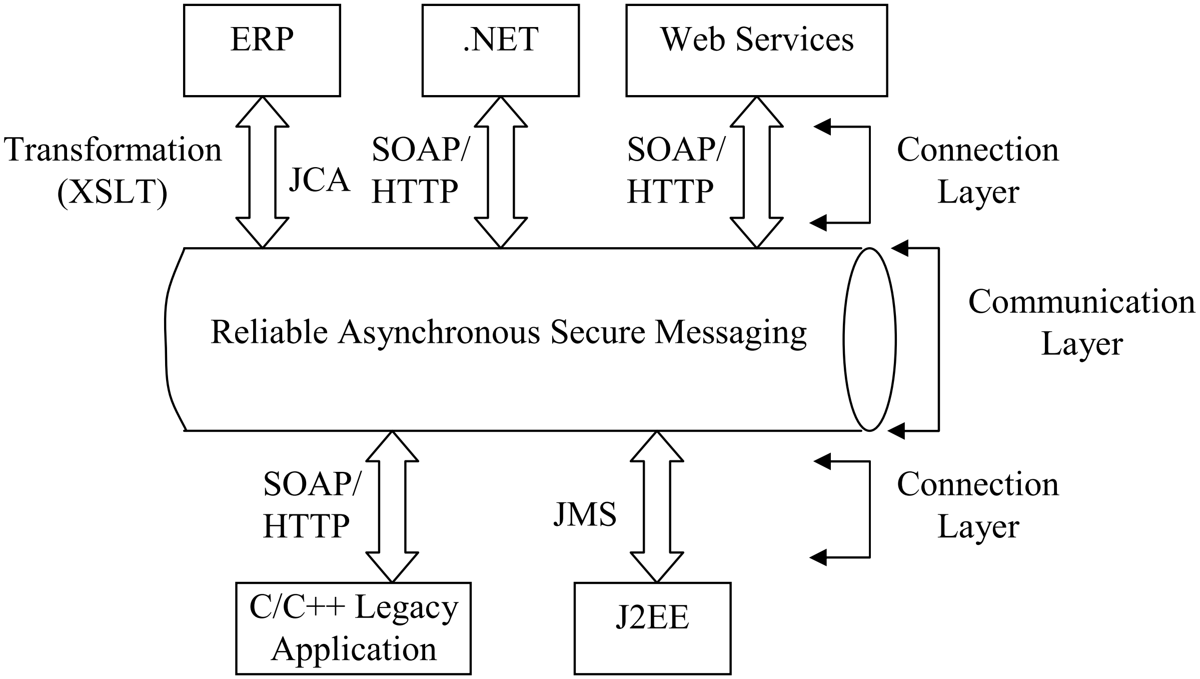
What capabilities should an ESB have?

- Service end-point connectivity
- Transformation
- Routing
- Extensibility to support future needs

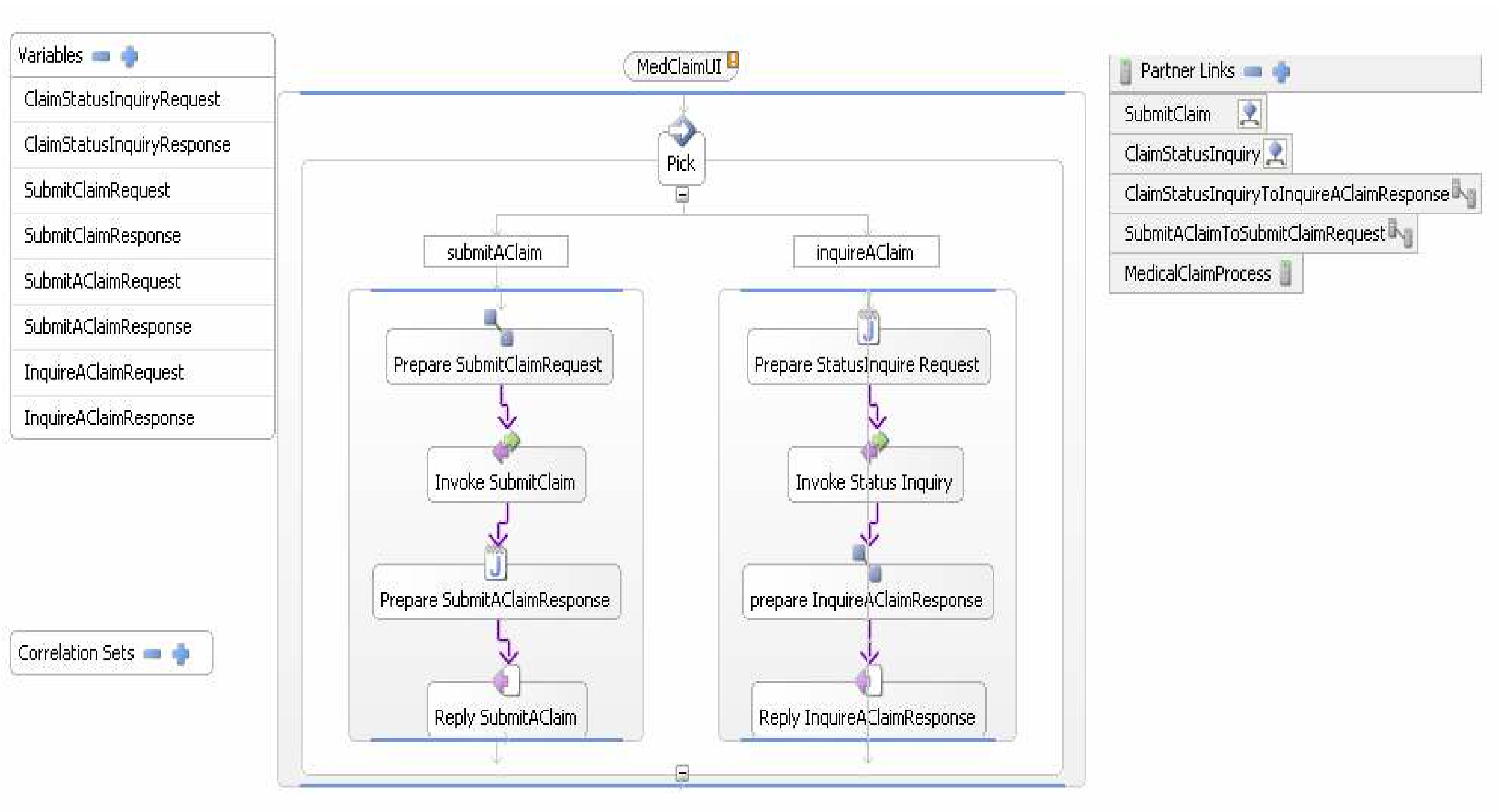
ESB (cont'd)

- The Enterprise Service Bus (ESB) provides key capabilities to enable ***Enterprise*** SOA.
- Provides an effective approach to service orchestration, application data synchronization, and business activity monitoring. ESBs offer the following key features:
  - **Web services**: support for SOAP, WSDL and UDDI, as well as emerging standards such as WS-Reliable Messaging and WS-Security
  - **Messaging**: asynchronous store-and-forward delivery with multiple qualities of service
  - **Data transformation**: XML to XML
  - **Content-based routing**: publish and subscribe routing across multiple types of sources and destinations
  - **Platform-neutral**: connect to any technology in the enterprise, e.g. Java, .Net, mainframes, and databases

- Advanced ESBs typically offer additional value-added features, including:
  - **Adapters**: enable connectivity into packaged and custom enterprise applications
  - **Distributed query engine**: enables the creation of data services from heterogeneous data sources
  - **Service orchestration engine**: supports long-running (stateful) and short-running (stateless) processes
  - **Application development tools:** allows the rapid creation of user-facing applications
  - **Presentation services**: enables the creation of personalized portals that aggregate services from multiple sources

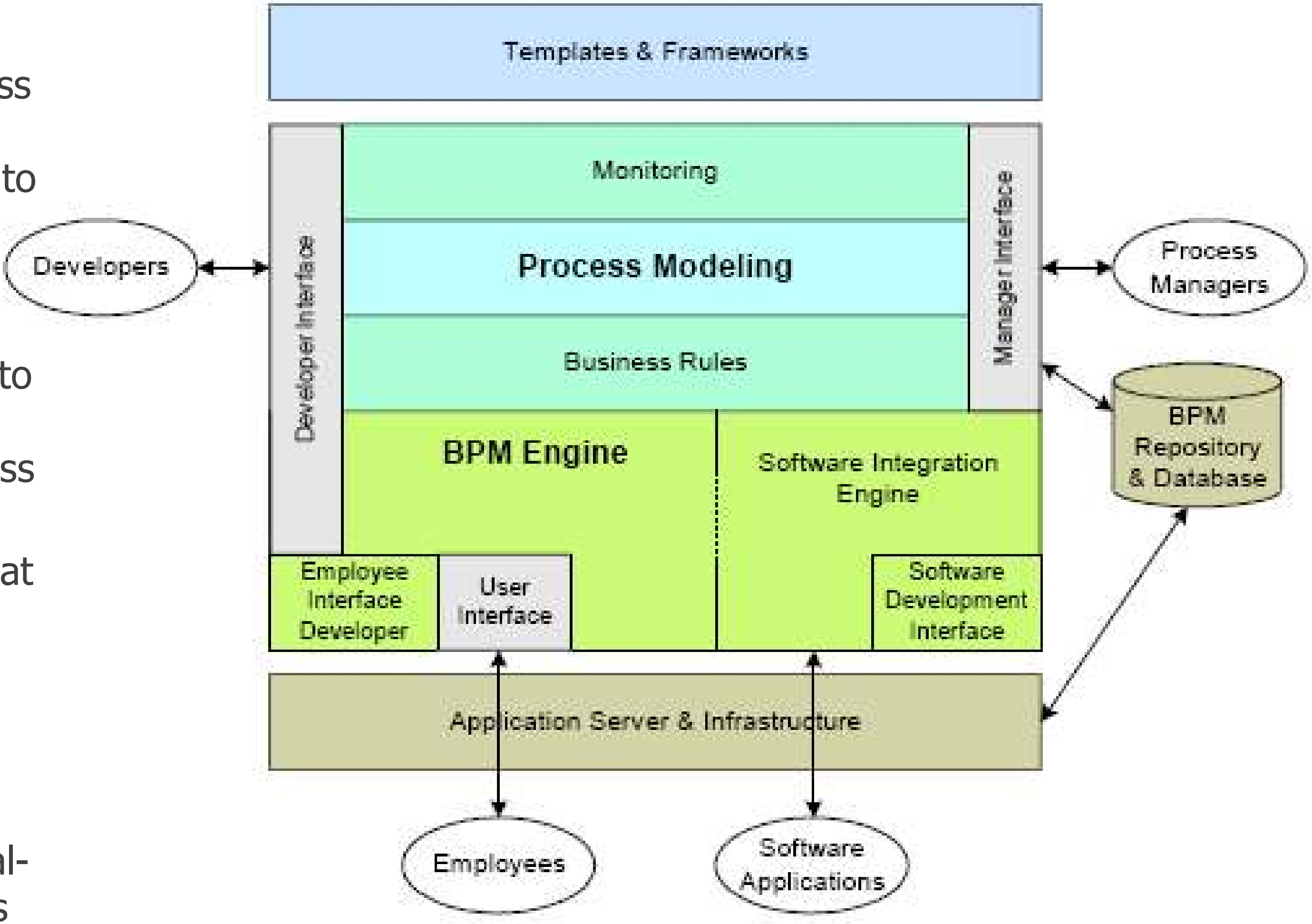


Business Process Execution Language (BPEL)



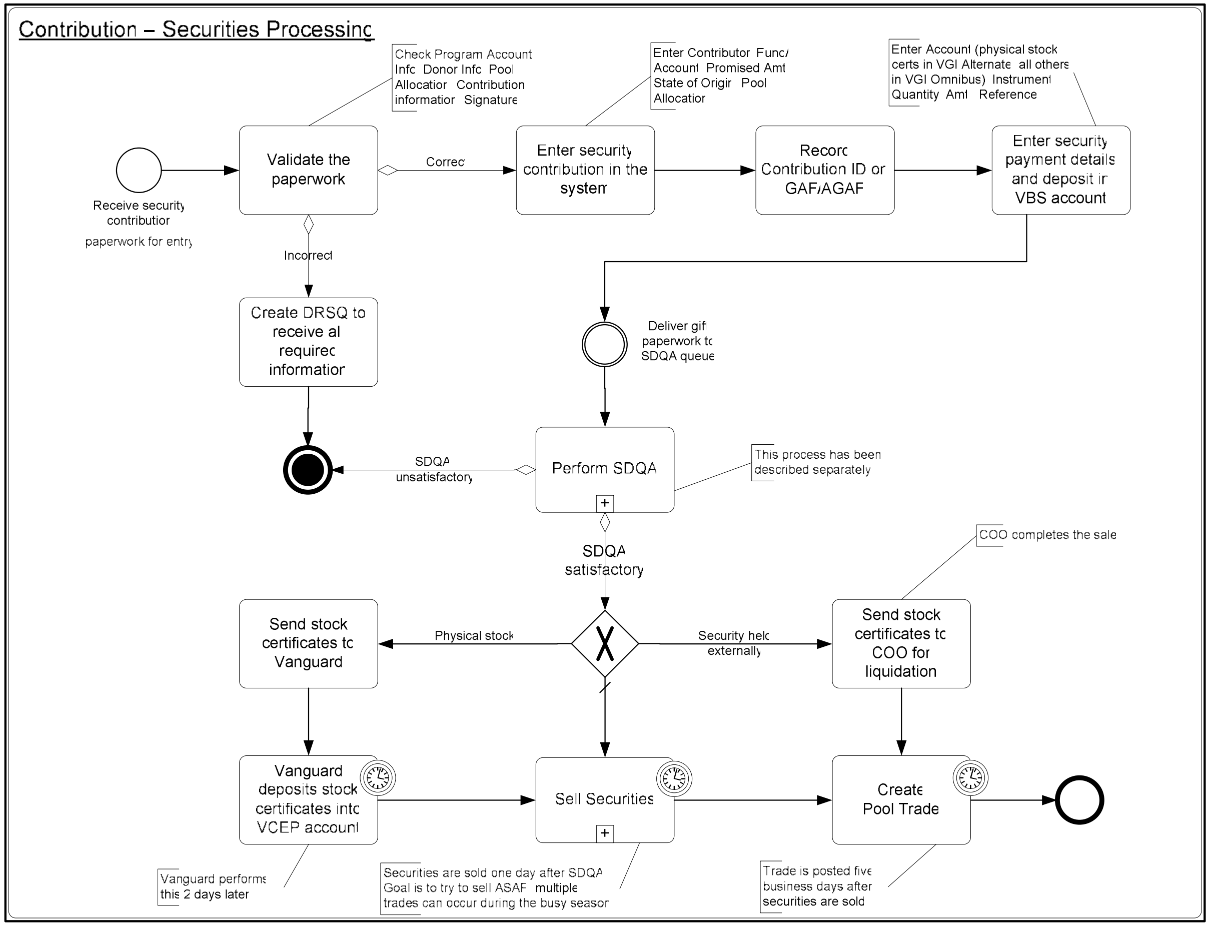
Workflow and BPM

- Integrated design and development environment
- Graphical tools to design, model and simulate a process under various conditions in preparation for deployment to production.
- Central Process Repository with version control functionality and the ability to capture core attributes associated with each business process.
- Process execution engine that manages the execution of business processes and maintains the state of each invoked process.
- Business Activity Monitoring (BAM) tools that provide real-time metrics of process KPIs and Dashboards.

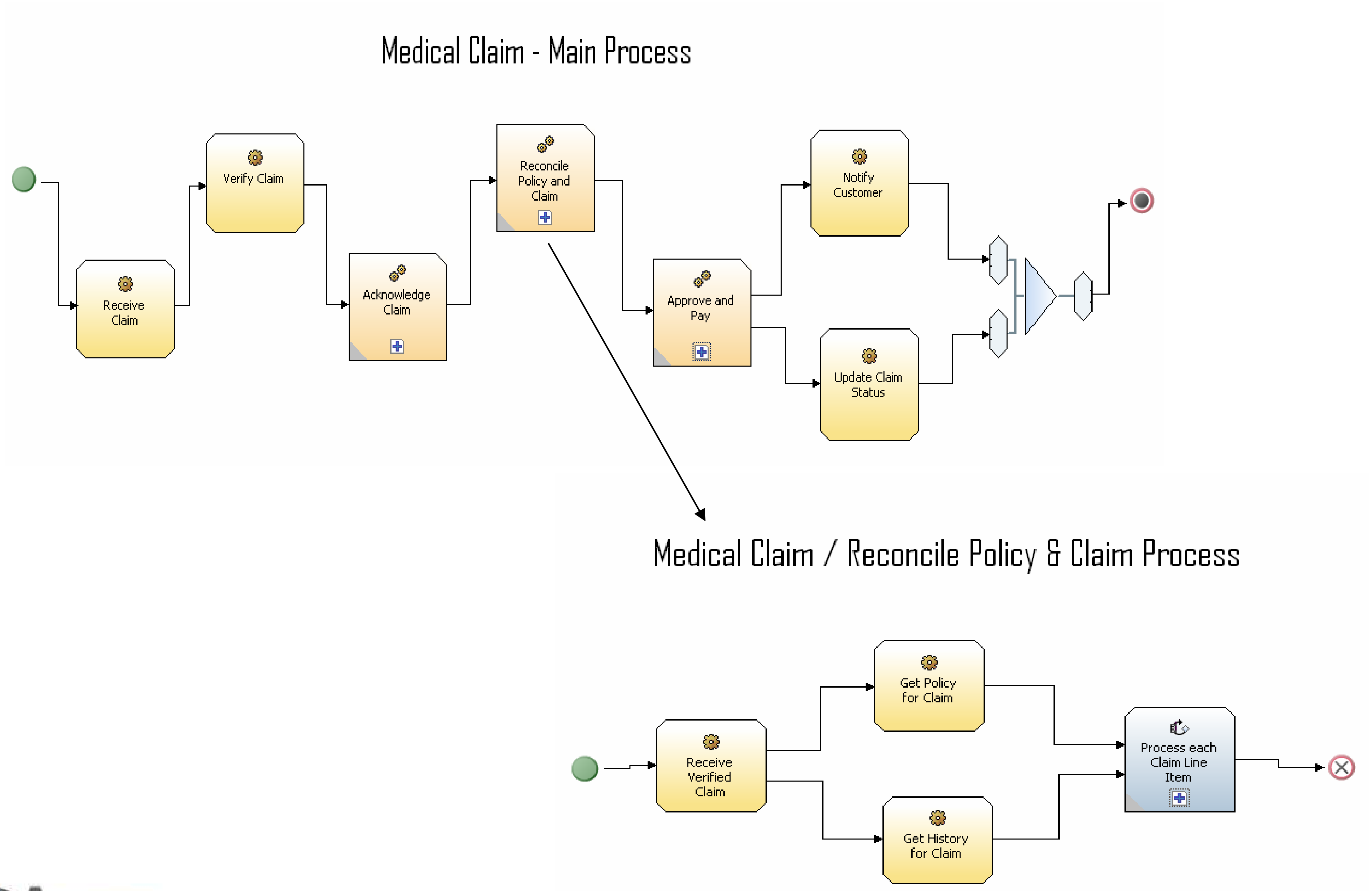


Workflow and BPM (cont’d)

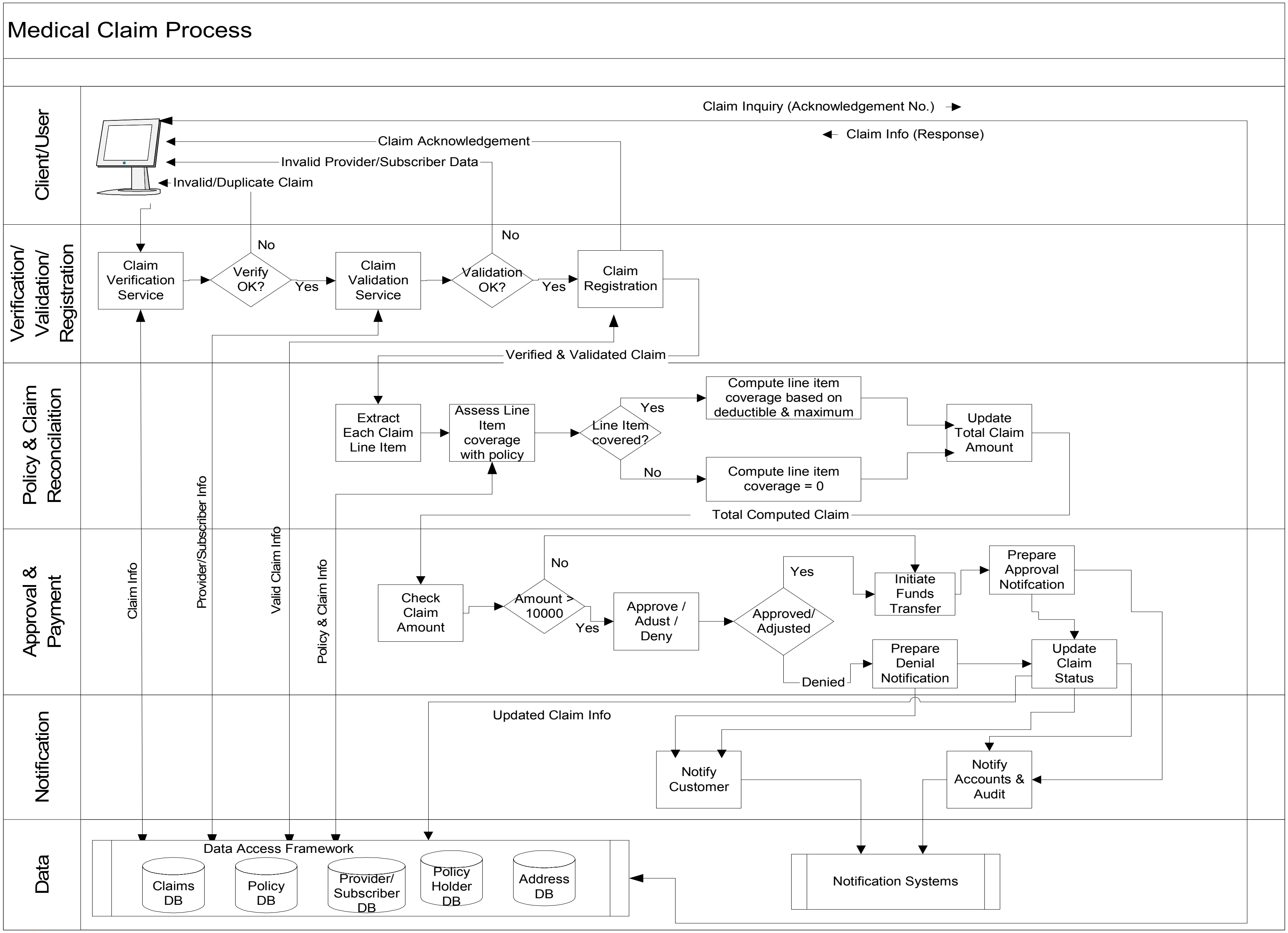
- Integrated design and development environment
- Graphical tools to design, model and simulate a process under various conditions in preparation for deployment to production.
- Central Process Repository with version control functionality and the ability to capture core attributes associated with each business process.
- Process execution engine that manages the execution of business processes and maintains the state of each invoked process.
- Business Activity Monitoring (BAM) tools that provide real-time metrics of process KPIs and Dashboards.


Contribution – Securities Processinc
Check Program Account Infc Donor Infc Pool Allocatior Contribution informatior Signature
Enter Contributor Func/ Account Promised Amt State of Origir Pool Allocatior
Enter Account (physical stock certs in VGI Alternate all others in VGI Omnibus) Instrument Quantity Amt Reference
Receive security contributior paperwork for entry
Validate the paperwork
Correct
Enter security contribution in the systerr
Recorc Contribution ID or GAF/AGAF
Enter security payment details and deposit ir VBS account
Incorrect
Create DRSQ to receive al requirec informatior
Deliver gift paperwork tc SDQA queue
SDQA unsatisfactory
Perform SDQA
This process has been described separately
SDQA satisfactory
COO completes the sale
Send stock certificates tc Vanguard
Physical stock
Security helc externally
Send stock certificates tc COO for liquidation
Vanguard deposits stock certificates intc VCEP accounl
Sell Securities
Create Pool Trade
Vanguard performs this 2 days later
Securities are sold one day after SDQA Goal is to try to sell ASAP multiple trades can occur during the busy season
Trade is posted five business days after securities are solc

BPM: *Business Process Model (Level 0, 1)*



Workflow and BPM (cont'd): *Business Process Diagramming*



Business Activity Monitoring (BAM)

MONITOR is the window into your business processes

What does it do?

- **Business Activity Monitoring**
  - Shows the minute-by-minute operation of the business on custom dashboards
  - Send alerts to allow the business to react to out-of-line conditions
  - Actively tracks business events through their execution across the value chain

What does it mean to you?

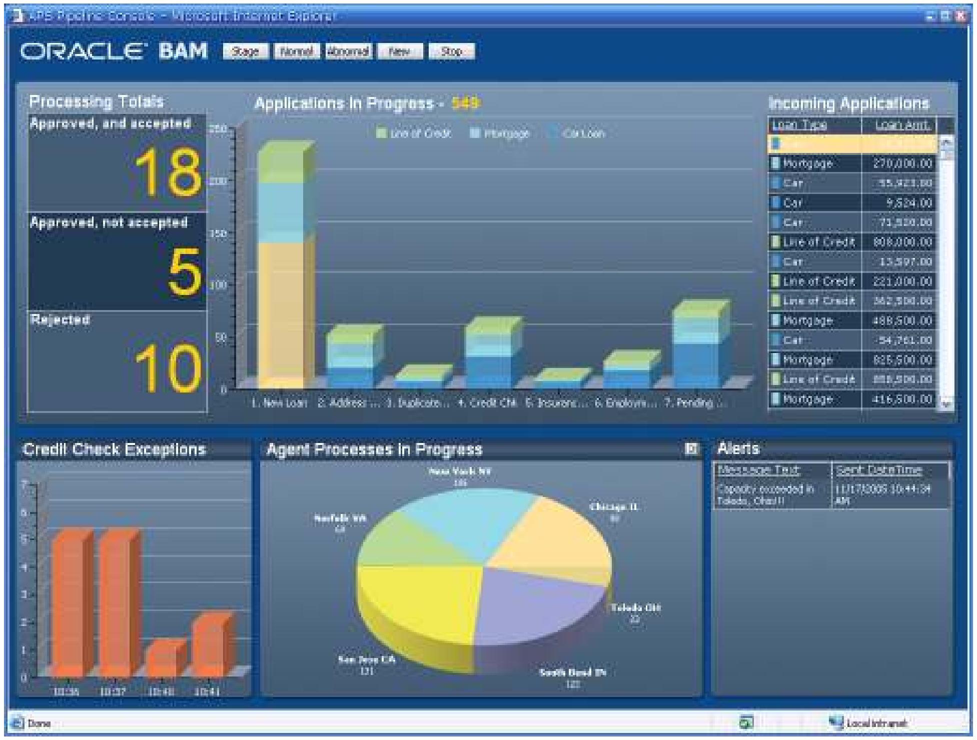
Line of Business:

- Business-tailored dashboards
- Up-to-the-minute status of orders, claims, invoices and other metrics
- Reports on key performance indicators

CIO:

- Allows optimized, near-real time decision making
- Alerts to ensure service levels
- Business measure to justify and confirm investments

Business Activity Monitoring (BAM)

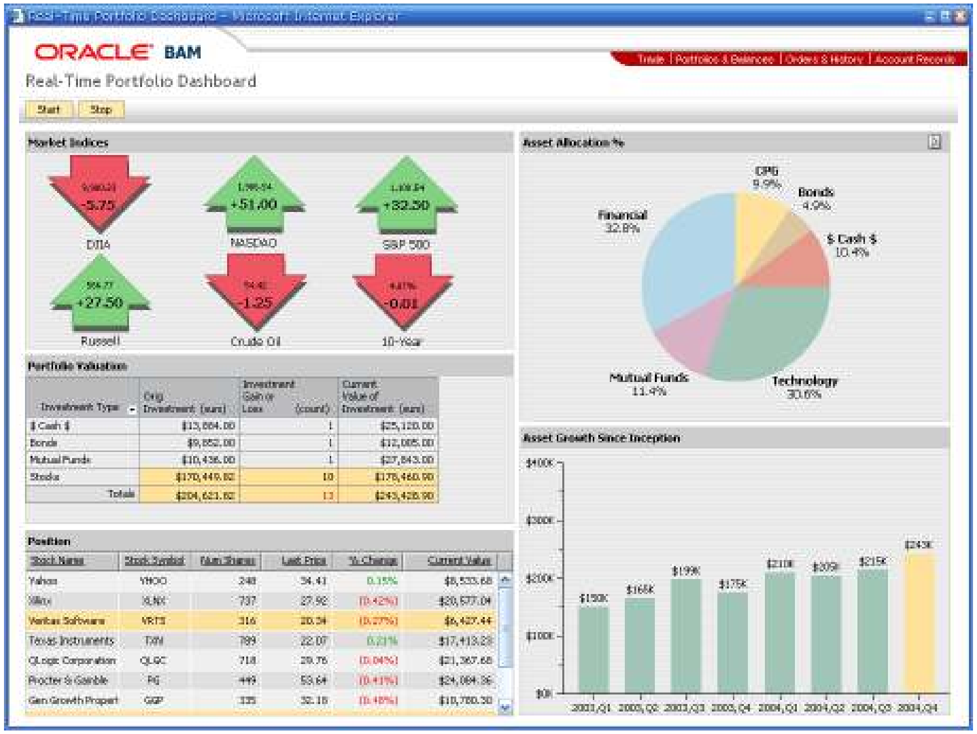


Incoming Applications

Loan Type	Loan Amt.
[illegible]	[illegible]
Mortgage	270,000.00
Car	55,923.00
Car	9,524.00
Car	71,520.00
Line of Credit	808,000.00
Car	13,597.00
Line of Credit	221,000.00
Line of Credit	362,500.00
Mortgage	488,500.00
Car	54,761.00
Mortgage	825,500.00
Line of Credit	858,500.00
Mortgage	416,500.00

Alerts

Message Text	Sent DateTime
Capacity exceeded in Toledo, Ohio!!!	11/17/2005 10:44:34 AM



Portfolio Valuation

Investment Type	Orig Investment (sum)	Investment Gain or Loss (count)	Current Value of Investment (sum)
$ Cash $	$13,884.00	1	$25,120.00
Bonds	$9,852.00	1	$12,005.00
Mutual Funds	$10,436.00	1	$27,843.00
Stocks	$170,449.82	10	$178,460.90
Totals	$204,621.82	13	$243,428.90

Position

Stock Name	Stock Symbol	Num Shares	Last Price	% Change	Current Value
Yahoo	YHOO	248	34.41	0.15%	$8,533.68
Xilinx	XLNX	737	27.92	(0.42%)	$20,577.04
Veritas Software	VRTS	316	20.34	(0.27%)	$6,427.44
Texas Instruments	TXN	789	22.07	0.21%	$17,413.23
QLogic Corporation	QLGC	718	29.76	(0.04%)	$21,367.68
Procter & Gamble	PG	449	53.64	(0.41%)	$24,084.36
Gen Growth Propert	GGP	335	32.18	(0.48%)	$10,780.30

SOA Registry and Repository

- **Registry:** A registry stores information about services in an SOA. It includes information that other participants can look up to find out the location of the service and what it does.
  - A registry may also include information about policies that are applied to the service, such as security requirements, quality of service commitments and billing
- **Repository:** Stores all services-related artifacts in the enterprise-wide SOA implementation. The repository should also provide cooperation capabilities (the ability to search, modify, etc.) to all of the SOA stakeholders

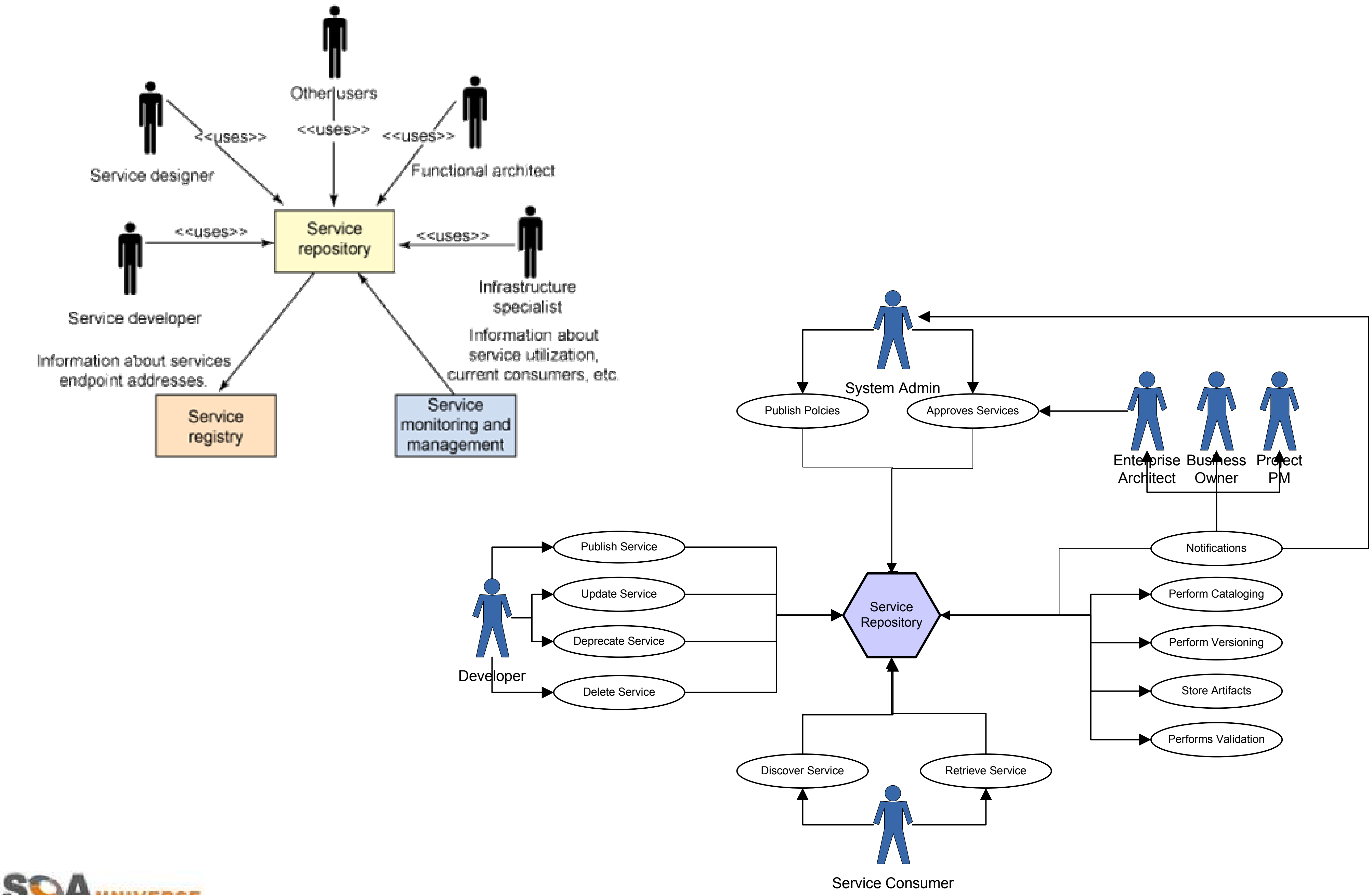
What’s the Difference Between a SOA Registry and Repository?

- The Repository contains all of the design and development artifacts of services that the design tools may need at design and build time
- The Registry contains a subset of this information that is required at runtime binding.

- The Service Repository is optimized for store large amounts of assets and to enable a large user population to make ad-hoc queries to find these assets
- The Service Registry is optimized for runtime lookups of services endpoint addresses

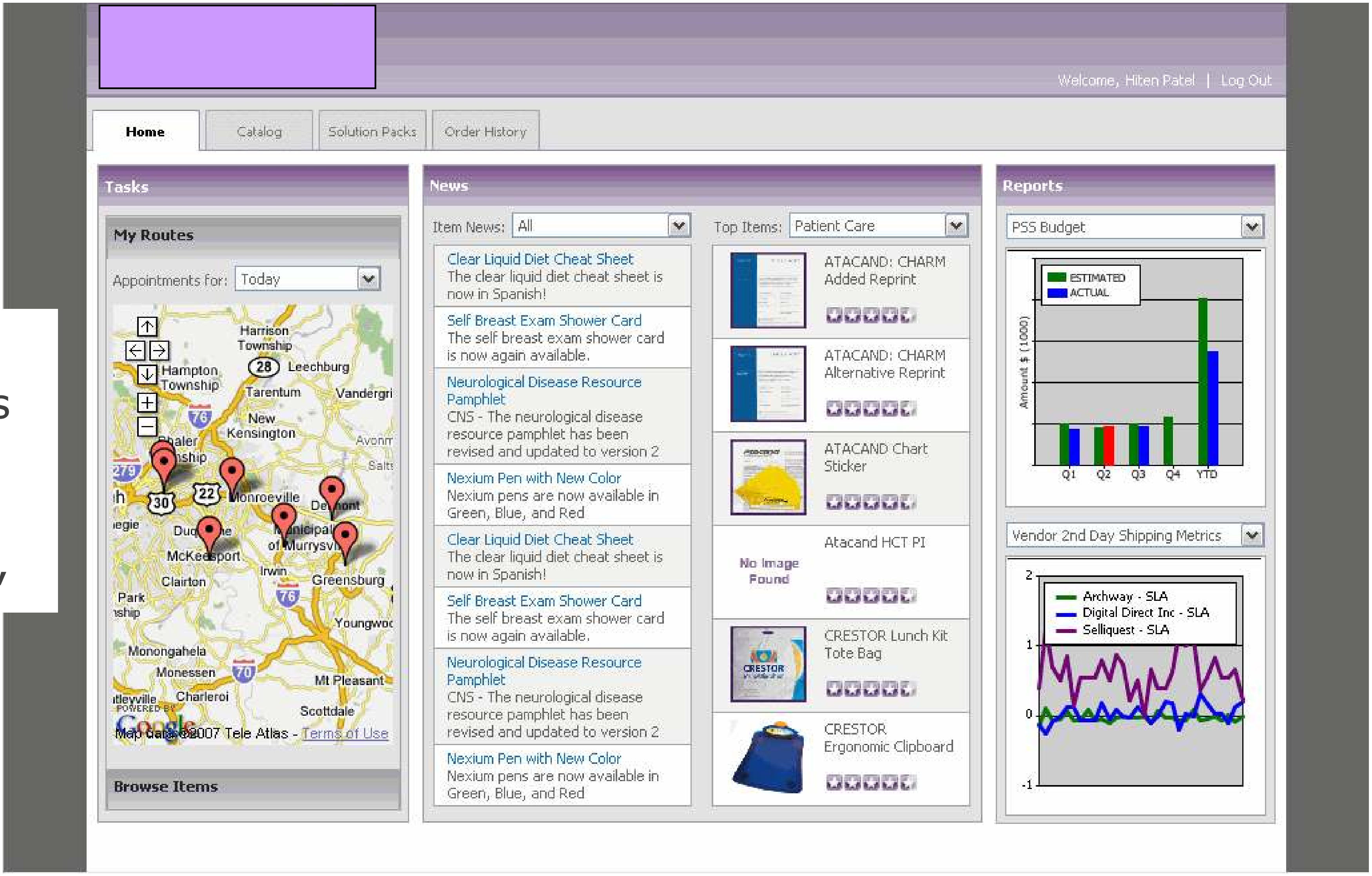
- Access to the Repository takes place within the enterprise boundaries
- The Registry often needs to be accessed from within and from the outside of these boundaries.

Examples of Registry / Repository Use Cases



Web2.0 and SOA: Even more Possibilities..

- AJAX
- Mash-Ups
- New UI Delivery (Gadgets, Mobile)



Welcome, Hiten Patel | Log Out

Home | Catalog | Solution Packs | Order History

Tasks

My Routes

Appointments for: Today

Browse Items

News

Item News: All

Clear Liquid Diet Cheat Sheet
The clear liquid diet cheat sheet is now in Spanish!

Self Breast Exam Shower Card
The self breast exam shower card is now again available.

Neurological Disease Resource Pamphlet
CNS - The neurological disease resource pamphlet has been revised and updated to version 2

Nexium Pen with New Color
Nexium pens are now available in Green, Blue, and Red

Clear Liquid Diet Cheat Sheet
The clear liquid diet cheat sheet is now in Spanish!

Self Breast Exam Shower Card
The self breast exam shower card is now again available.

Neurological Disease Resource Pamphlet
CNS - The neurological disease resource pamphlet has been revised and updated to version 2

Nexium Pen with New Color
Nexium pens are now available in Green, Blue, and Red

Top Items: Patient Care

ATACAND: CHARM Added Reprint

ATACAND: CHARM Alternative Reprint

ATACAND Chart Sticker

No Image Found
Atacand HCT PI

CRESTOR Lunch Kit Tote Bag

CRESTOR Ergonomic Clipboard

Reports

PSS Budget

Vendor 2nd Day Shipping Metrics

SOA Technical Architecture: Discussion Questions

- What Technologies have your Development Teams adopted?
- What are the common benefits observed in adoption of WS, ESB, BPEL, BPM, BAM, and Service Registry and Repository?
- Has Business Process become a key part of your Thinking and Operations? If not, what are the biggest challenges?
- Has your Enterprise adopted a best-of-breed approach, or an consolidated vendor approach?
- What are your Experiences with and Plans for Web2.0?

SOA UNIVERSE
SOA Knowledge in Action

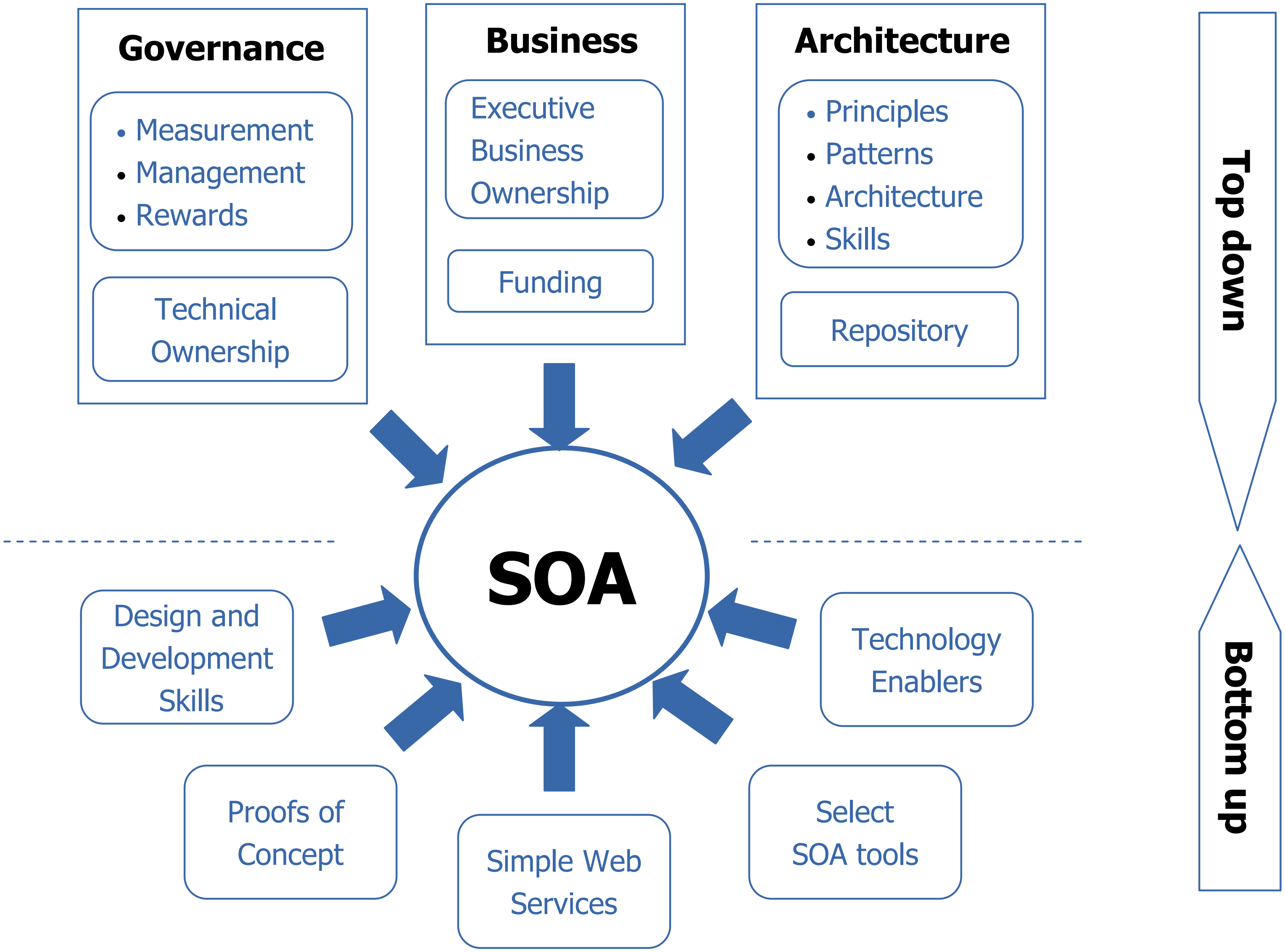
HURWITZ
& ASSOCIATES

liquidhub

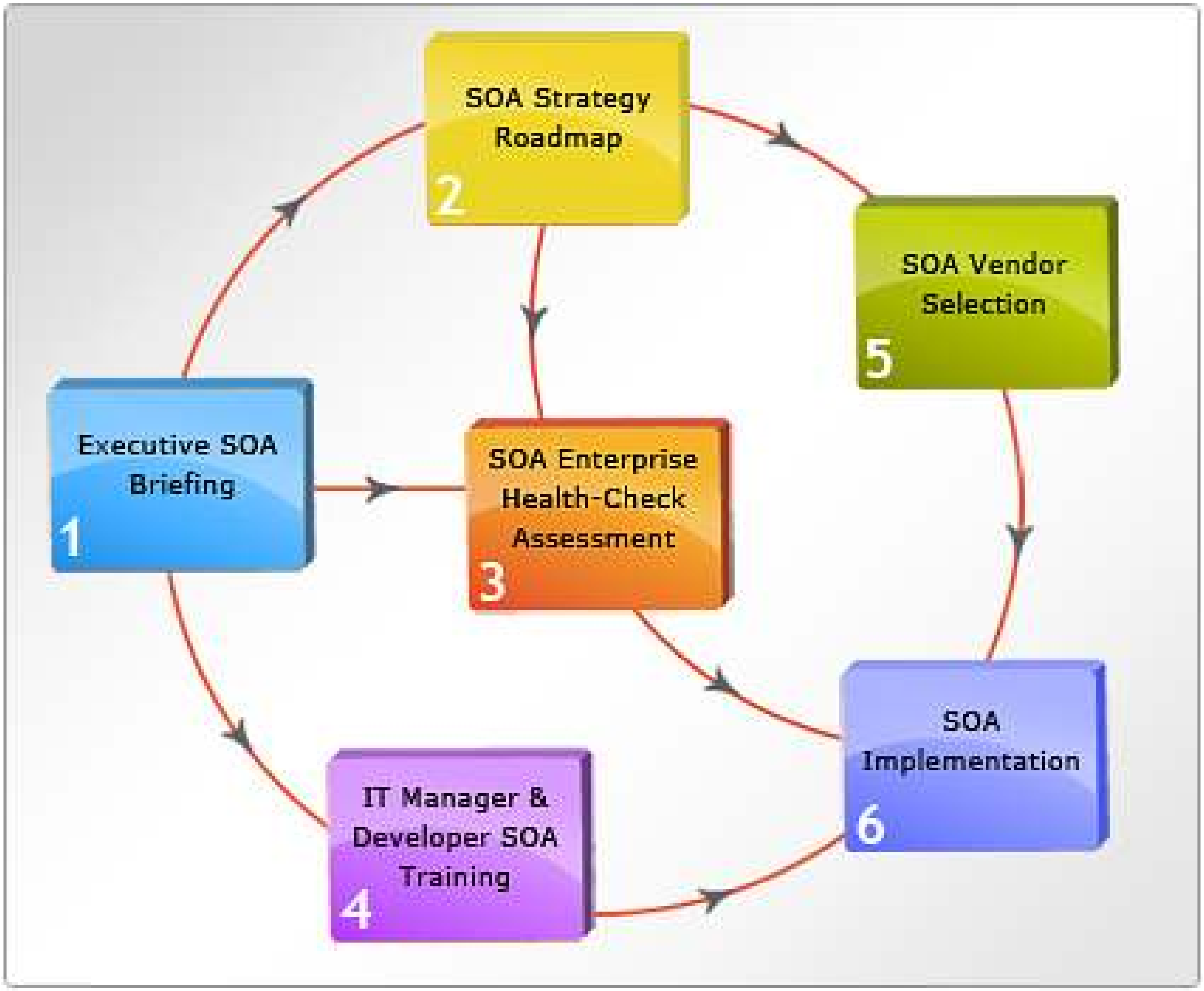
Section IV: Getting Started

What's Our Approach? *Top-Down, Bottom-Up, Middle-out, Hybrid?*

"Top Down" and "Bottom Up" considerations need to be balanced



Specific (Finite) Areas to Consider



Best Practice: *SOA Transition Planning*

- Define SOA Within Your Own Organization
  - Surprisingly, one of the greatest obstacles to adopting SOA is a lack of understanding of what actually constitutes a service oriented architecture. Without a clear vision of what SOA is in the abstract, you will be in no position to contribute to or even assess the merits of a transition plan.
- Invest in an Impact Analysis Before Developing the Transition Plan
  - In order to assess the feasibility of a transition to SOA, you first need to estimate the real-world impact such a migration will have.
  - SOA's reach is broad. This type of research effort tends to include in-depth assessments of current and upcoming development tools, infrastructure requirements, skill-set and training requirements, proposed new middleware, changes to organizational processes, changes to security models and policies, and a list of recommendations associated with architecture, custom standards, and project management approaches.
  - Keep in mind that if you decide to postpone your transition, much of this analysis work may lose its value.
- Set the scope of the transition
  - It's not uncommon for an SOA transition plan to apply only to a subset of an organization's technical environment
- Expect Evolution to be Part of the Migration
  - WS-* standards are volatile; products that implement these standards will undergo continuous refinements
- Use Speculative Analysis (6 to 12 months) to Build Toward a Future State

Best Practice: Create a Services Blueprint for Transformation

Partner & Supplier Interaction

- Brokers
- Trust Accounting
- External Advisors
- Transaction Clearing-houses

Fund Accounting

Cash Management
- Manage Cash

Pricing
- Price/Value Investments
- Correct Pricing Error

Securities Accounting
- Conduct Securities Lending
- Perform Clearing & Settlement

Analysis & Product Development

Performance Analysis
- Product/Service Success Analytics

Service Development
- R&D
- Product Lifecycle

Record Keeping

Process Information Requests
- Generate Reports
- Generate Statements
- Generate Confirm/ Notifications

Process Client Transactions
- Manage Assets
- Manage Account Balance
- Administer Installment Payment
- Rebalance Assets
- Calculate Benefits
- Convert New Plan
- Bill Fees
- Process Credit/Margin
- Calculate Annuity Payments
- Administer Annuitization Payment
- Manage Loans
- Manage Trust Assets
- Manage Trust Income & Disbursements
- Process Corporate Actions
- Withhold Taxes
- Purge/Archive Records
- Manage Client Payments
- Prepare Excess Refund
- Prepare Pass-Through Dividend
- Manage Brokerage Orders

Advisory Services

Financial Planning
- Create Financial Plan
- Manage Financial Plan

Account Management

Manage Accounts
- Setup/Maintain Person
- Setup/Maintain Account
- Setup/Maintain DC/DB Plan
- Setup/Maintain Trust

Manage Information Requests
- Provide Information
- Provide Personalized Performance Data

Prepare Client Transactions
- Prepare Purchase
- Prepare Redemption
- Prepare Exchange
- Prepare Buy
- Prepare Sell
- Exercise Options
- Prepare Contribution
- Prepare Withdrawal
- Prepare Rollover
- Perform Adjustments
- Administer Installment Setup
- Administer Annuitization Setup
- Administer Loan
- Prepare Asset/ Account Transfer
- Prepare Plan Level Transfer
- Prepare 1035 Exchange
- Terminate Person
- Prepare Death Claim

Investment Management

Investment Strategies Management
- Manage Portfolios
- Replicate Indexes
- Manage Order Routing and Execution
- Monitor Performance

Customer Relationship Management

Marketing
- Analyze/Understand Client
- Perform Market Research/Analysis
- Create/Modify Products/Services
- Educate Client
- Prepare Communications

Customer Service
- Call Center Services
- Customer Lifecycle
- Customer Segment Management

Sales Force Automation
- Campaign Management
- Contact Management
- Sales Goal Performance Management/Dashboard

Channels

- Kiosk/POS
- Call Center/IVR
- Web
- Client
- Mobile
- Fax
- Paper

Customer Segments

- Retirement Client
- Brokerage Client
- Endowment Client
- Trust Client
- Defined Contribution Client
- Defined Benefit Client

Management & Operations

Account Reconciliation
- Account
- Reconciliation
- Reconcile Checks
- Reconcile Client Accounts
- Reconcile Transfer Agency Accounts
- Reconcile Custody Bank Accounts
- Reconcile Omnibus Accounts

Compliance
- Monitor Investment Compliance
- Monitor Client Compliance
- Audit Dividend & Capital Gain Disbursements

Client Control Reporting
- Process As-of Transactions
- Provide Tax Services

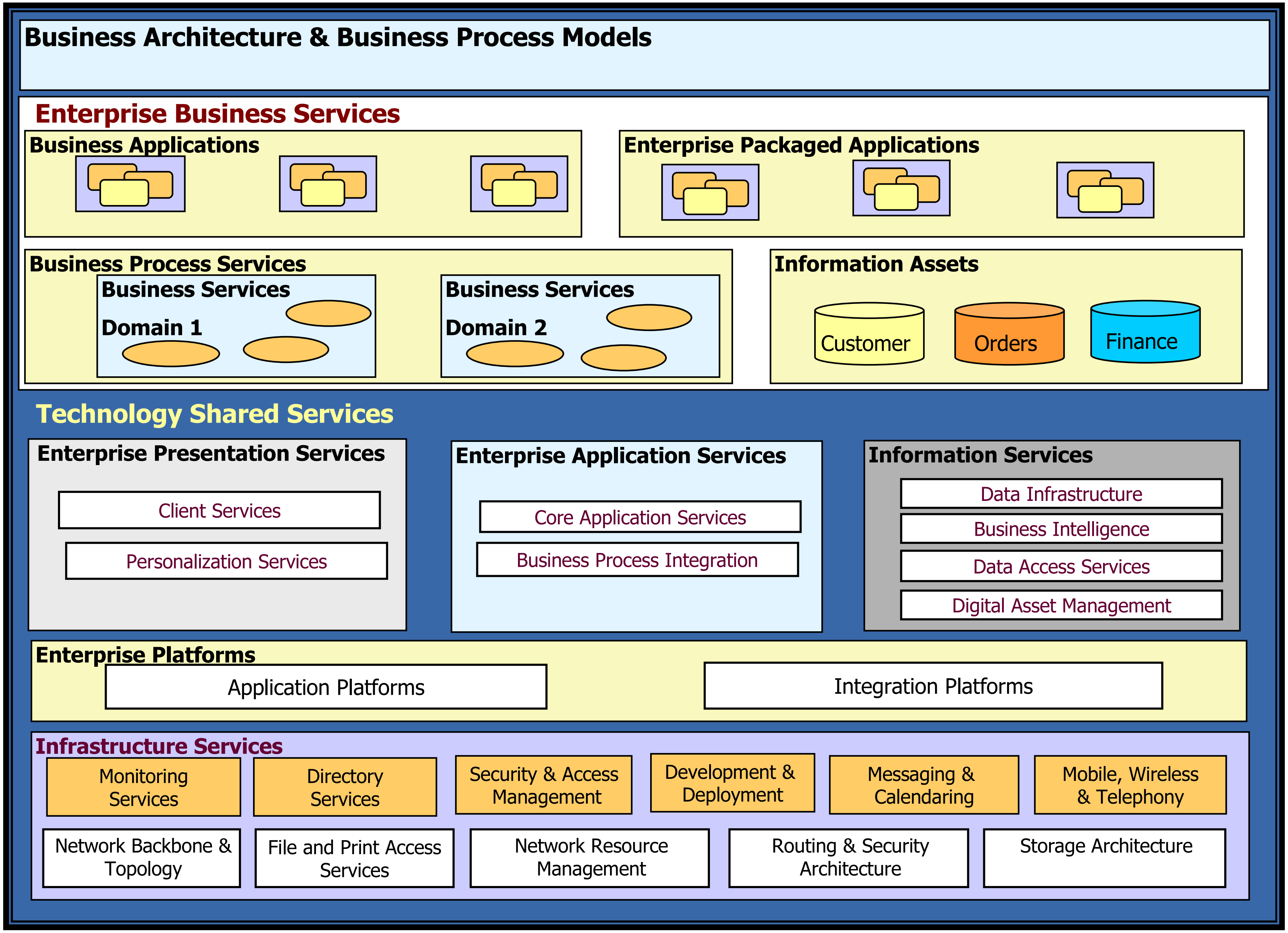
Money Movement
- Move Money

Inventory Management
- Manage Literature Inventory
- Fulfill Literature Requests

Financial Management
- Perform Corporate Budgeting
- Provide Executive Information
- Execute Monthly/Yearly Financials
- Perform AP/AR
- Issue Payroll
- Track Assets
- Prepare Compliance Reporting

Human Resources
- Hire/Retain/Release Employees
- Provide Career Management
- Provide Work/Life Initiatives
- Prepare Compliance Reporting

Best Practice: *Create an ESA Reference Model*



Business Architecture & Business Process Models

Enterprise Business Services

Business Applications

Enterprise Packaged Applications

Business Process Services

Business Services Domain 1

Business Services Domain 2

Information Assets

Technology Shared Services

Enterprise Presentation Services
- Client Services
- Personalization Services

Enterprise Application Services
- Core Application Services
- Business Process Integration

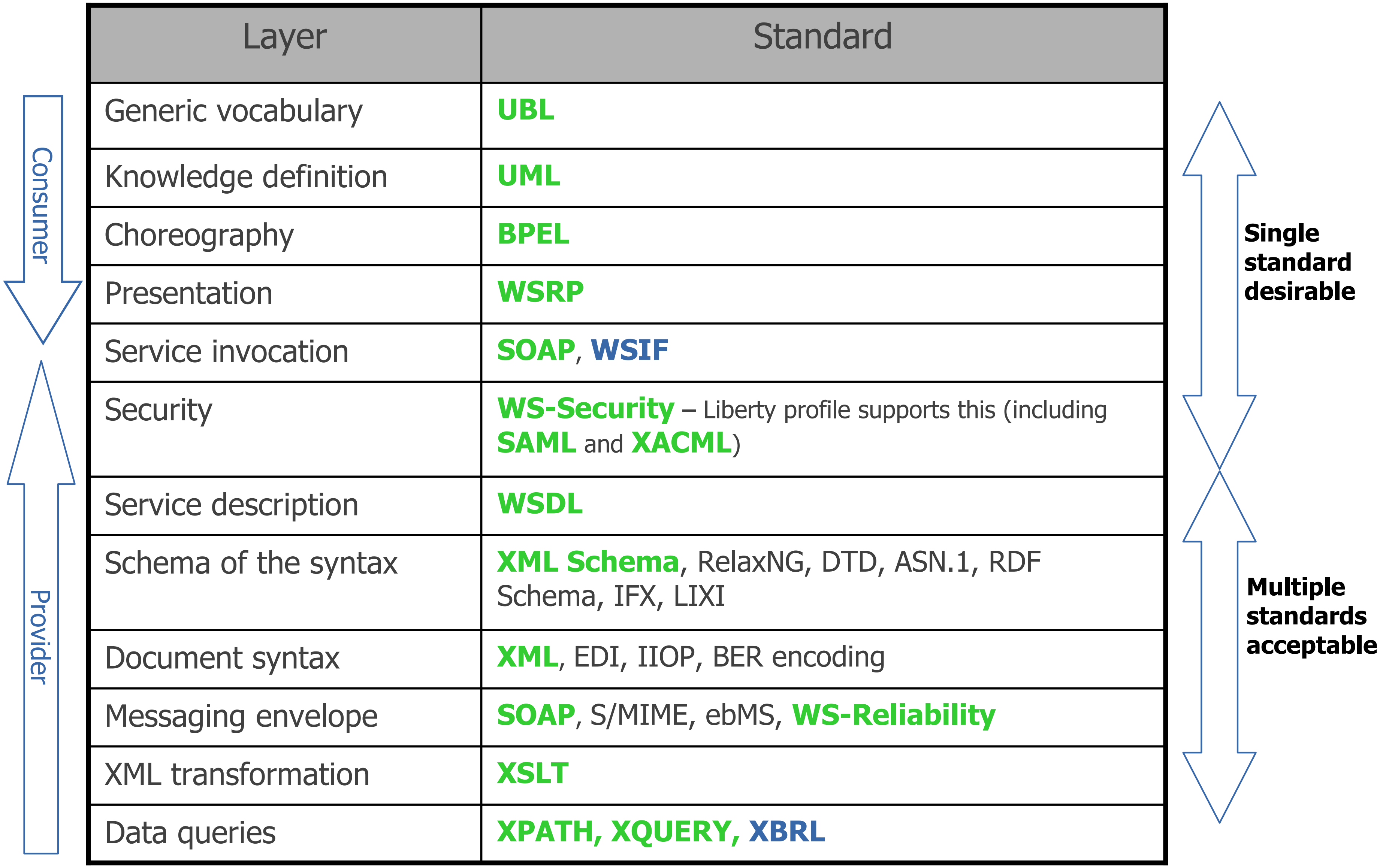
Information Services
- Data Infrastructure
- Business Intelligence
- Data Access Services
- Digital Asset Management

Enterprise Platforms
- Application Platforms
- Integration Platforms

Infrastructure Services
- Monitoring Services
- Directory Services
- Security & Access Management
- Development & Deployment
- Messaging & Calendaring
- Mobile, Wireless & Telephony
- Network Backbone & Topology
- File and Print Access Services
- Network Resource Management
- Routing & Security Architecture
- Storage Architecture

SOA UNIVERSE SOA Knowledge in Action

Best Practice: *Create & Publish an SOA Standards Stack*



Consumer

Provider

Layer	Standard
Generic vocabulary	**UBL**
Knowledge definition	**UML**
Choreography	**BPEL**
Presentation	**WSRP**
Service invocation	**SOAP**, **WSIF**
Security	**WS-Security** – Liberty profile supports this (including **SAML** and **XACML**)
Service description	**WSDL**
Schema of the syntax	**XML Schema**, RelaxNG, DTD, ASN.1, RDF Schema, IFX, LIXI
Document syntax	**XML**, EDI, IIOP, BER encoding
Messaging envelope	**SOAP**, S/MIME, ebMS, **WS-Reliability**
XML transformation	**XSLT**
Data queries	**XPATH, XQUERY, XBRL**

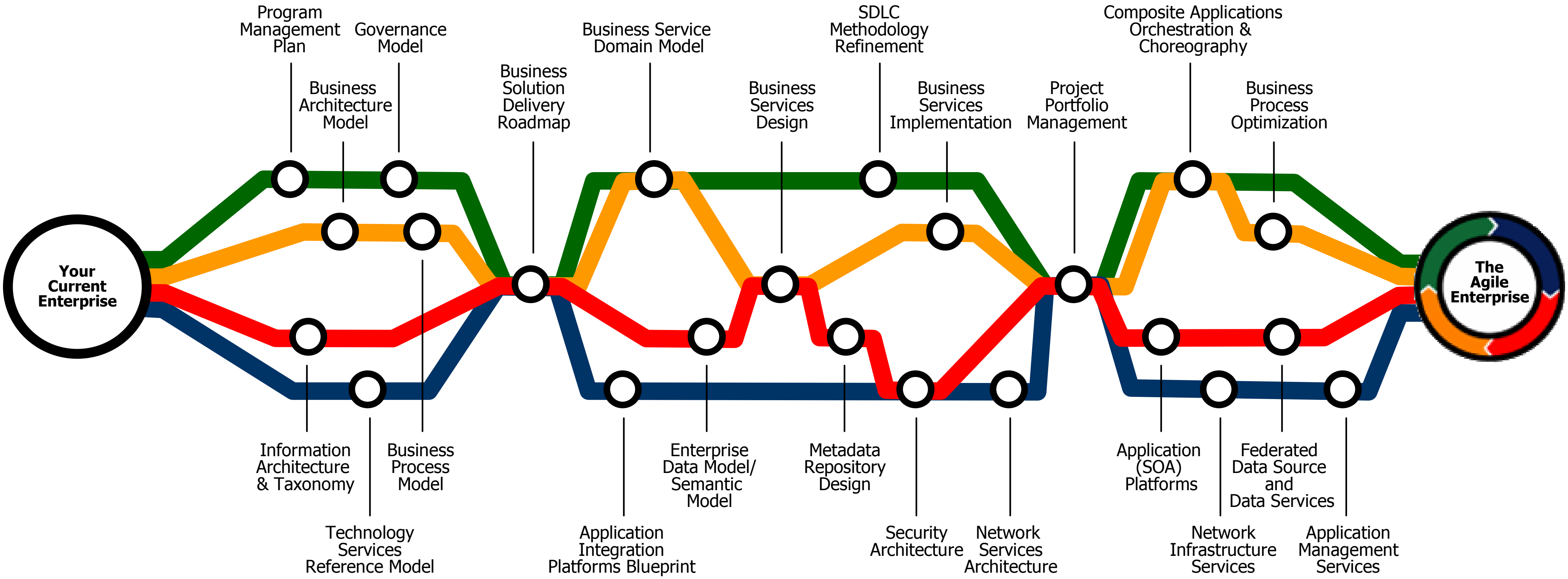
Single standard desirable

Multiple standards acceptable

Common Pitfalls of Adopting SOA

- Technical-Only Approach in the Absence of Organizational and Business Alignment
  - Socialization
  - Education
- Building Service Oriented Architectures like traditional distributed architectures
  - Problems:
    - proliferation of RPC-style service descriptions (leading to increased volumes of fine-grained message exchanges)
    - inhibiting the adoption of features provided by WS-* specifications
    - Further entrenchment of synchronous communication patterns
    - Creation of hybrid or non-standardized services
- Not creating a Transition Plan
  - Migration needs to happen at the technical, process and organization levels to avoid ad-hoc adoption
- Not Standardizing SOA
  - Like any other architecture, SOA requires the creation and enforcement of design standards
- Failing to Create an XML Foundation Architecture
  - SOA requires standardizing how core XML technologies are used to represent, validate and process corporate data
- Failing to Account for SOA Performance Requirements
  - As message-based communication increases, processing latency can be an issue
- Lack of Proper SOA Security Model
  - Secure Sockets Layer (SSL) is not the technology of choice for SOA; the need for message-level security implies that the WS-Security Framework is optimal
- Failure to Understand Prudent Use of Standards
  - Web Services Interoperability (WS-I): Basic Profile and Basic Security Profile

SOA *Truly* is an Enterprise Transformation



Your Current Enterprise

Program Management Plan

Business Architecture Model

Governance Model

Information Architecture & Taxonomy

Technology Services Reference Model

Business Process Model

Business Solution Delivery Roadmap

Business Service Domain Model

Application Integration Platforms Blueprint

Enterprise Data Model/ Semantic Model

Business Services Design

SDLC Methodology Refinement

Metadata Repository Design

Business Services Implementation

Security Architecture

Network Services Architecture

Project Portfolio Management

Composite Applications Orchestration & Choreography

Business Process Optimization

Application (SOA) Platforms

Network Infrastructure Services

Federated Data Source and Data Services

Application Management Services

The Agile Enterprise

Phases

Envision (Strategy and Plan)	**Engineer** (Architect/Design)	**Execute** (Develop/Implement)

Work Streams

🟩 **Architecture Governance**

🟧 **Enterprise Business Services**

🟥 **Information and Data**

🟦 **Technology Shared Services**

Getting Started: Discussion Questions

- Have you taken a Top-Down, Bottom-Up, Middle-Out or Hybrid Approach to your SOA Activities?
- How have you chosen your initial SOA Projects?
- What has and hasn't worked within your approach?
- How do you gauge your Organization's readiness re: SOA Adoption?

SOA UNIVERSE
SOA Knowledge in Action

Section V: Thoughts on Operationalizing SOA

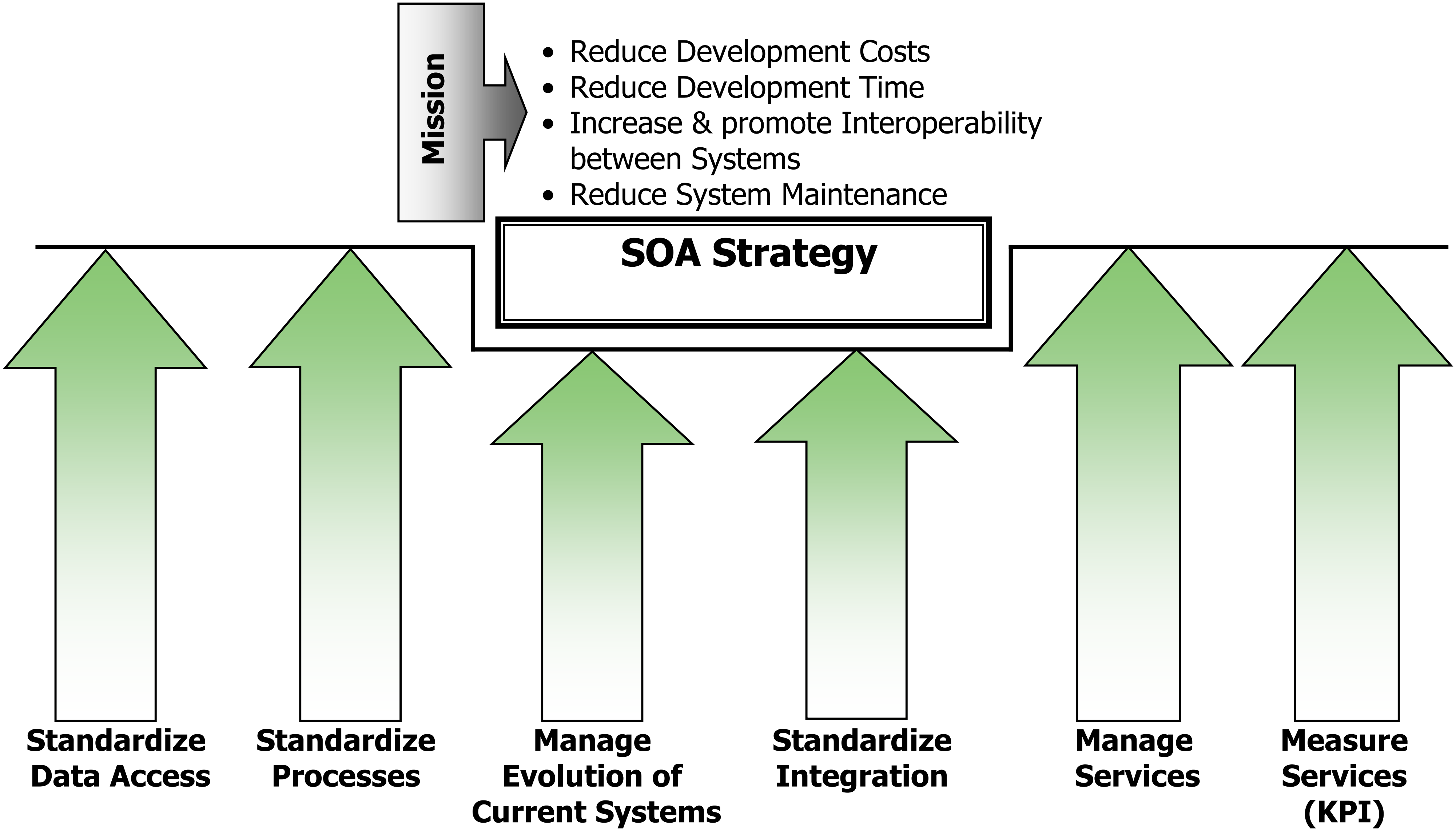
HURWITZ
& ASSOCIATES

liquidhub

Thoughts on Operationalizing

- Surviving (*well*) is Key
- Technical
  - KPIs and Measures
  - Performance
  - Capacity Planning
  - Simulation
  - Service Publishing / Approval
- Business
  - Funding (psst: We are all in “Sales”)
- Governance

Technical: How Do We Measure Success?



Mission

- Reduce Development Costs
- Reduce Development Time
- Increase & promote Interoperability between Systems
- Reduce System Maintenance

SOA Strategy

Standardize Data Access

Standardize Processes

Manage Evolution of Current Systems

Standardize Integration

Manage Services

Measure Services (KPI)

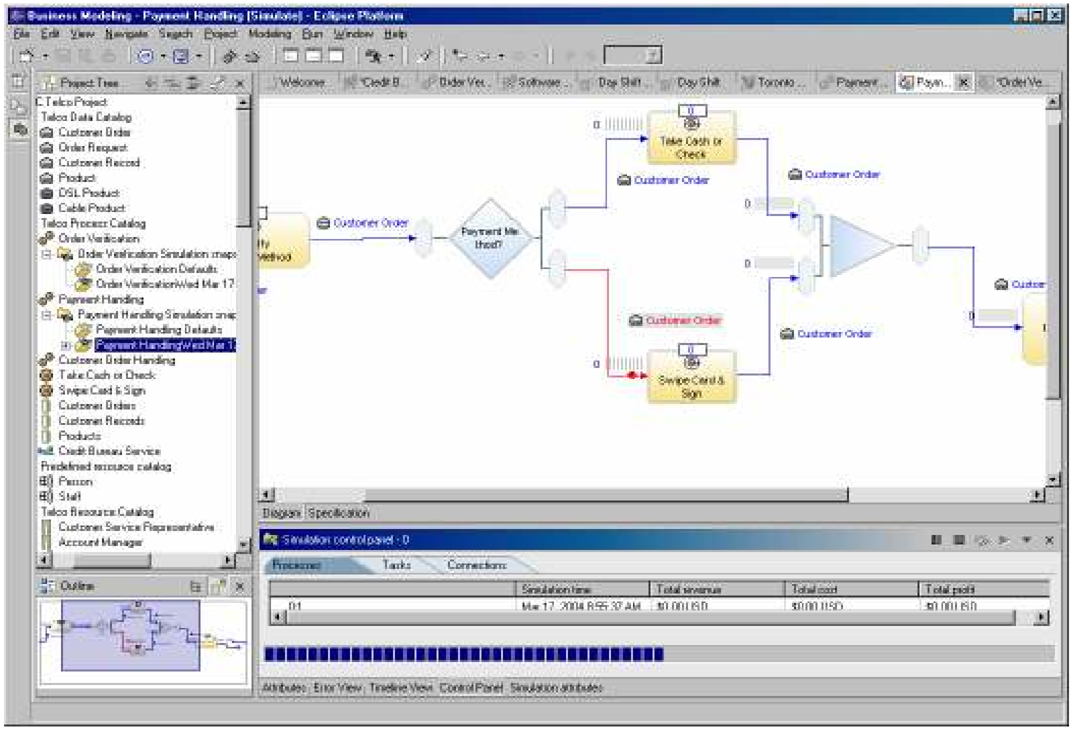
Technical: Performance and Capacity Planning

- Message-based communication in SOAs can, in fact, increase performance requirements when compared to RPC-style communication within traditional distributed architecture
  - XML processing-related performance challenges (Encryption, etc.)
  - Stress-testing of vendor supplied components (for XML, XSLT, SOAP, etc..)
  - Alternative processors, accelerators or other types of technology:
    - XML-binary Optimized Packaging (XOP)
    - SOA Message Transmission Optimization Mechanism (MTOM)
- Service Granularity is also crucial: Coarse-grained service interfaces and asynchronous messaging are emphasized when building Web Services
- There is no substitute for Good Design!

- Capacity Planning becomes Geometrically more difficult in SOA
  - CPU / Disk
  - “Container” Health and “Process Presence”
  - Application-Specific KPIs
  - Now add:
    - Unknown applications / users accessing my Service(s)
    - Unpredictable combinations/workloads

Technical: Simulation as a Reality Check

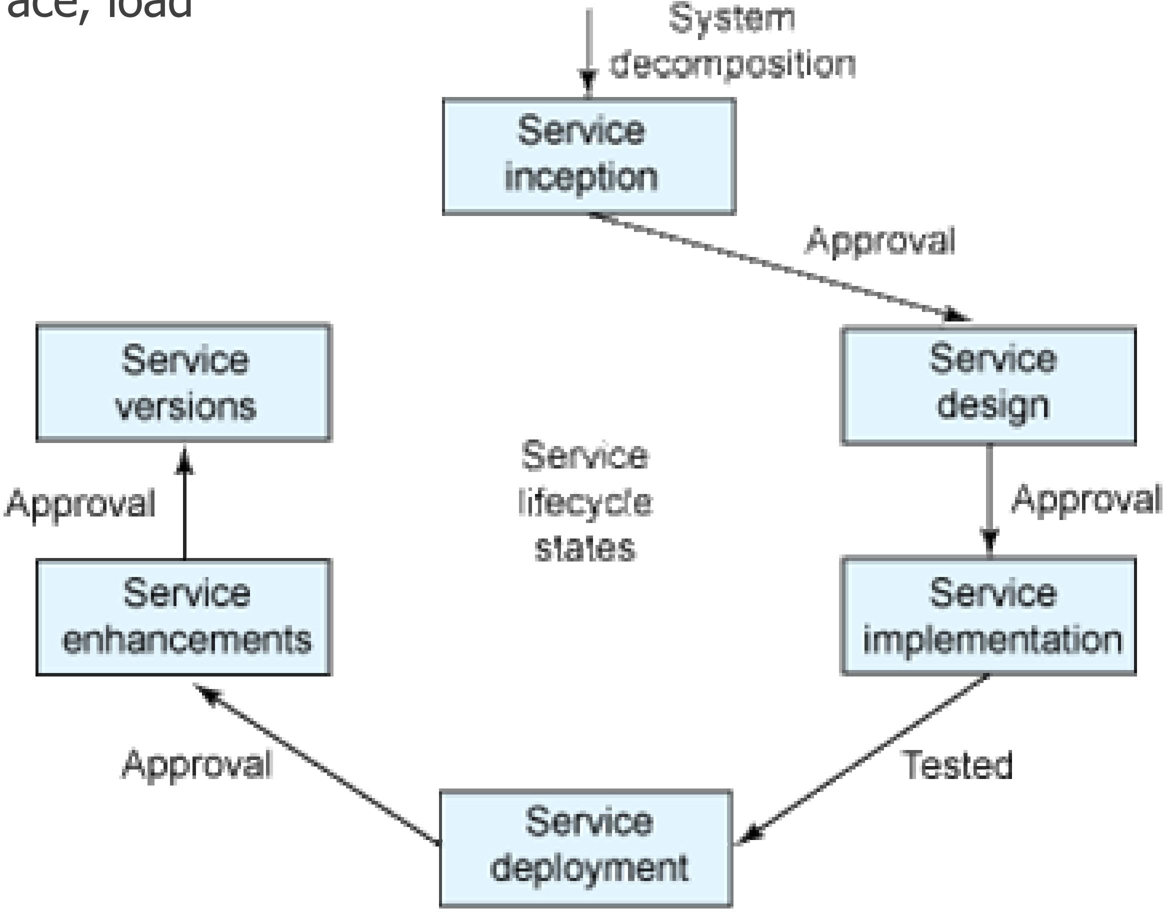
- Weighted average analysis provides a static, long-term view of the process; process simulation captures the shorter-term view
- Ability to model "what if" scenarios and compare results and replay a simulation of a process with changes to the model
- Sophisticated modeling and distribution for resources (individual and bulk), resource skills, resource allocations, cost, revenue and processing time
- Simulation output provides detailed information regarding resource utilization levels, as well as cost and cycle time calculations
- Powerful simulation engine supports conditional branching, steady-state model, run persistence, and multi-process concurrent simulation



	Simulation time	Total revenue	Total cost	Total profit
01	Mar 17, 2004 8:55:37 AM	$0.00 USD	$0.00 USD	$0.00 USD

Technical: Service Publishing / Approval

- Design and Testing Reviews are critical, as are published SLAs
  - Related artifacts are published as well (design docs, service interface, load testing results, etc).



Business: Operationalizing the Funding of SOA

- Who should bear the costs of services that span the enterprise?
  - Initial business Adopter
  - Corporate IT
  - Combination
- What are Approaches that can delivery on Reuse and Leverage promised by SOA?

Business: SOA Entrepreneur Model

Entrepreneur Model: Business Develops Services and “Sells” Back to Enterprise-at-Large

- IT-savvy Business Units (especially early adopters) are encouraged to perform initial Service development and meet immediate needs
- Services are “sold” back to the Enterprise to offset the Business Unit’s development costs
- Promotes wide applicability and rapid evolution
- However, higher risks including service duplication, granularity a too-low a level, and exposure of weaknesses in Enterprise Data Strategy

→Works best in Organizations that have deep IT skills embedded within the Business Units as well as an evolved Enterprise Data Strategy and strong Project Management disciplines

→Enables the fastest adoption of SOA within an Enterprise

Business: SOA Utility Model

CIO/CEO Identifies and Funds Keys Services

- Centralized Governance body collects business requirements, sets priority and sequence of Service Development, and provides funding through a Corporate budget
- First Services to be developed should be most widely used or have the most impact across business units.
- Or Governance body may choose to fund a small number of "incubator" Services to gain expertise in Service granularity or to gauge the impact of SOA on the SDLC

→Works best in Organizations with strong centralized development teams and credible Governance.

→Highly effective to show SOA progress and accountability to the Executive team and useful in educating re: benefits of the SOA strategy

→Measured (but slow) adoption of SOA principles

Business: SOA Partnership Model

IT and Business Jointly Fund Service Development

- IT and the Business meet in smaller, more nimble sub-teams to set strategy regarding Service granularity and design
- Rationalized and agreed-upon Services are jointly developed
- Based on internal chargeback mechanisms, the cost of Service development is shared between IT and the Business Units.
- Leads to tighter integration of the teams as well as shared responsibility and more effective resource utilization.

→Works best in Organizations with effective distributed development teams / matrixed organizations

→Ensures that IT and the Business are in "lock step" as SOA adoption unfolds

→Demands continual, close coordination, trust, and joint ownership from both parties

→"Reasonable" Speed

Funding Models: Common Themes

- Visible Governance Body: must oversee the overall SOA Strategy, and clearly define the Operating Model that an Organization will adopt (before SOA deployment begins).
- Governance Body could be only a lightweight clearinghouse, but must be the focal point where SOA activities are at least cataloged
- Funding / Chargeback Approach: An internal "Market Approach" regarding the SOA Funding is essential to provide with clear metrics regarding the total cost, and to ensure that resources are being spent wisely.
- Corporate Data Strategy: Initial design and development of Services needs to highly leverage the Organization's Data and Information Strategy (or will expose the lack thereof)

→ As SOA takes hold as both a Business and IT-led phenomenon, forward-thinking Organizations will take the opportunity to gain new insight into funding their development efforts *and* transform the Business / IT relationship

Governance 101

Authority

You, or you in the context of a team have to provide evidence to leadership or stakeholders that the actions and deliverables of a SDLC phase are in conformance with the expected requirements for that phase in order to ensure success.

The bottom-line is that the burden of success or failure for the delivery of that phase is upon you or your team.

Accountability

You are a decision-maker and enabler for those decisions to be realized.

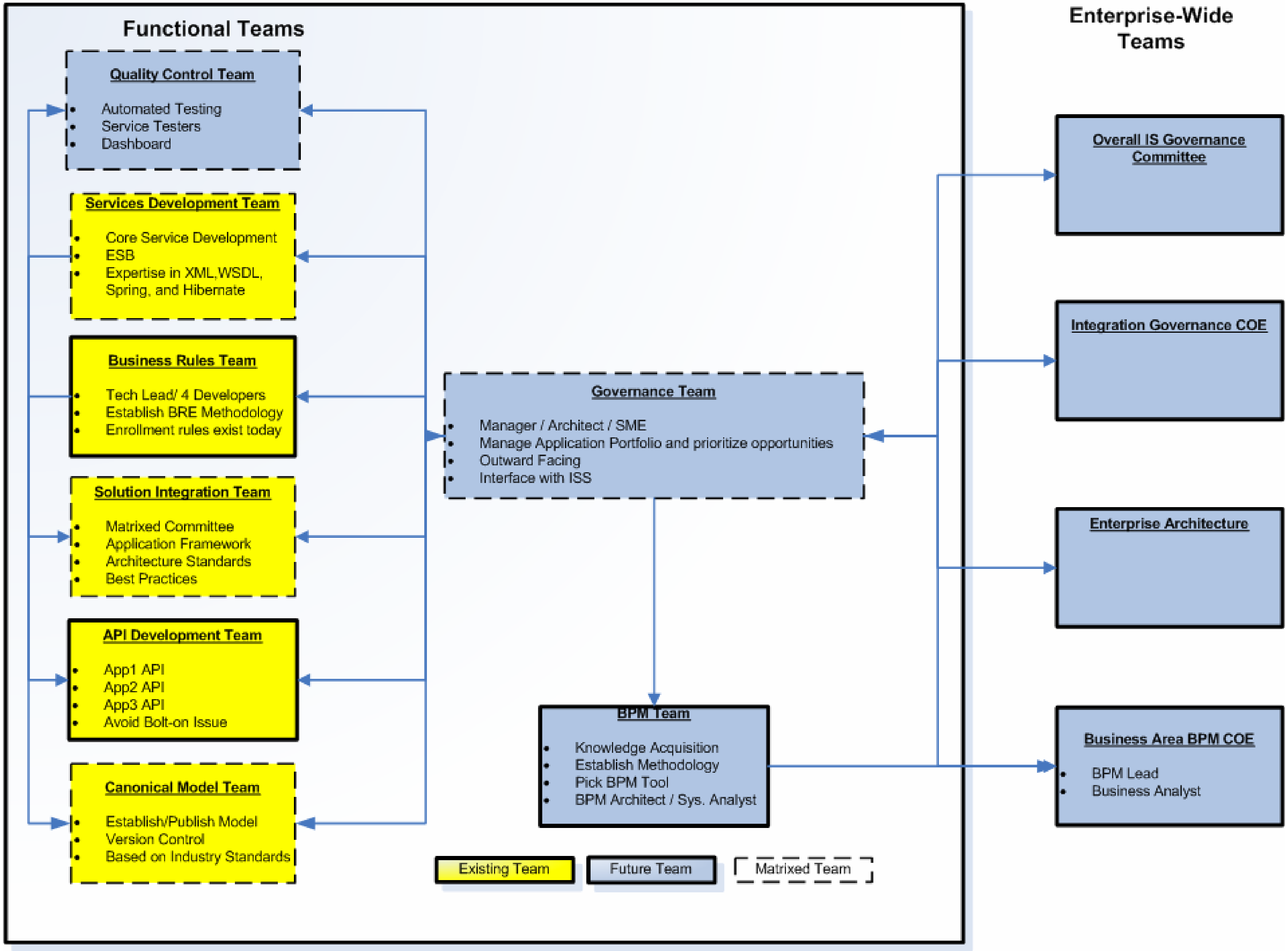
Bottom-line, based upon intelligent reasoning and experience, you can initiate actions that will cause effect and resolution.

Responsibility

You are held responsible when the actionable activities within specific phase of a SDLC can be traced back to you.

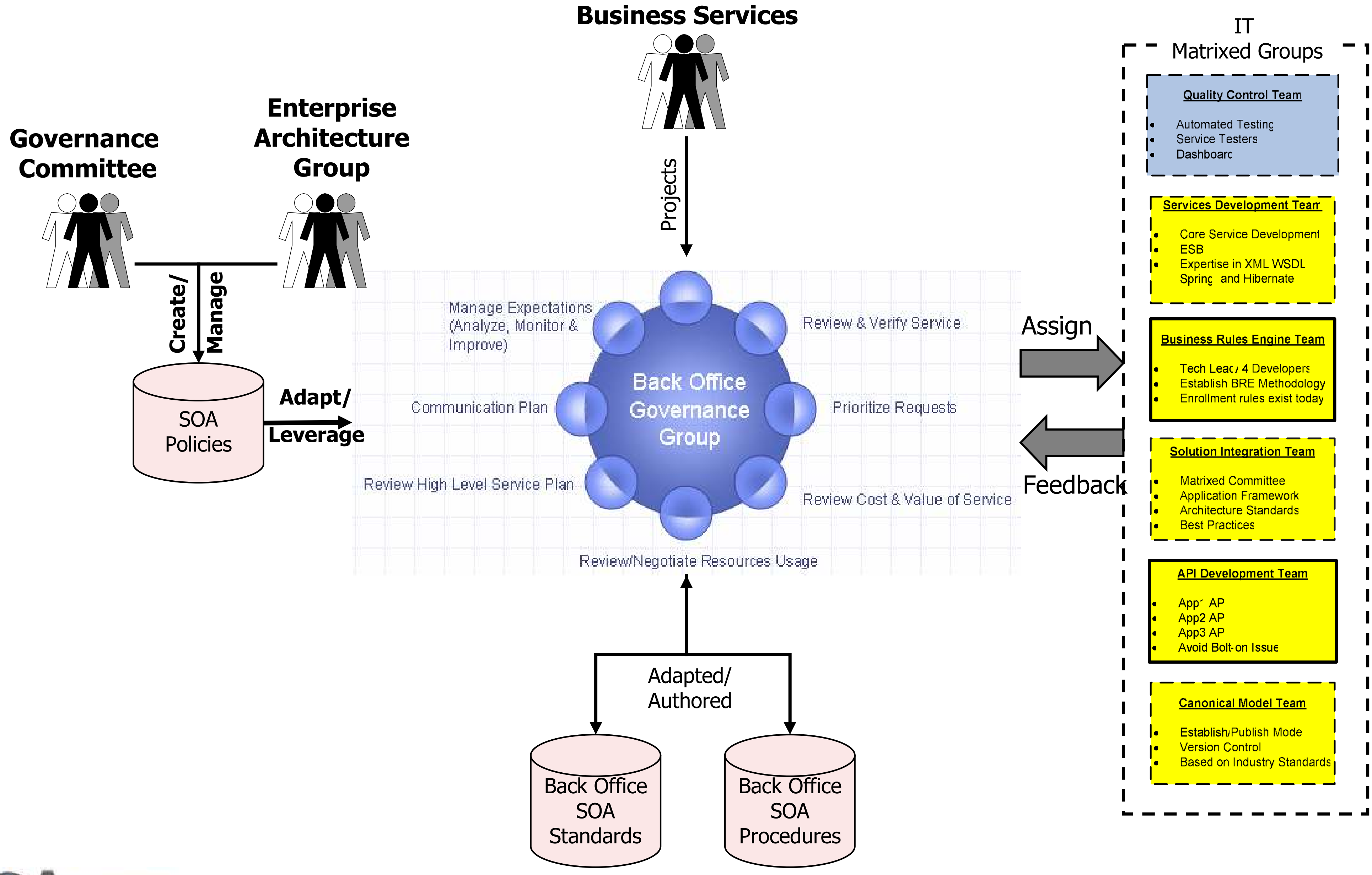
Bottom-line, there is tight-coupling between given a task to perform and executing that task.

Governance: An Example of a Federated Model



Cross-Functional team members from all departments are used on a project by project basis

Federated Governance in Action



Business Services

Projects

Governance Committee

Enterprise Architecture Group

Create/ Manage

SOA Policies

Adapt/ Leverage

Back Office Governance Group

Manage Expectations (Analyze, Monitor & Improve)

Review & Verify Service

Communication Plan

Prioritize Requests

Review High Level Service Plan

Review Cost & Value of Service

Review/Negotiate Resources Usage

Adapted/ Authored

Back Office SOA Standards

Back Office SOA Procedures

Assign

Feedback

IT Matrixed Groups

Quality Control Team

- Automated Testinc
- Service Testers
- Dashboarc

Services Development Tean

- Core Service Developmen
- ESB
- Expertise in XML WSDL Sprinc and Hibernate

Business Rules Engine Team

- Tech Leac/ 4 Developers
- Establish BRE Methodology
- Enrollment rules exist today

Solution Integration Team

- Matrixed Committee
- Application Framework
- Architecture Standards
- Best Practices

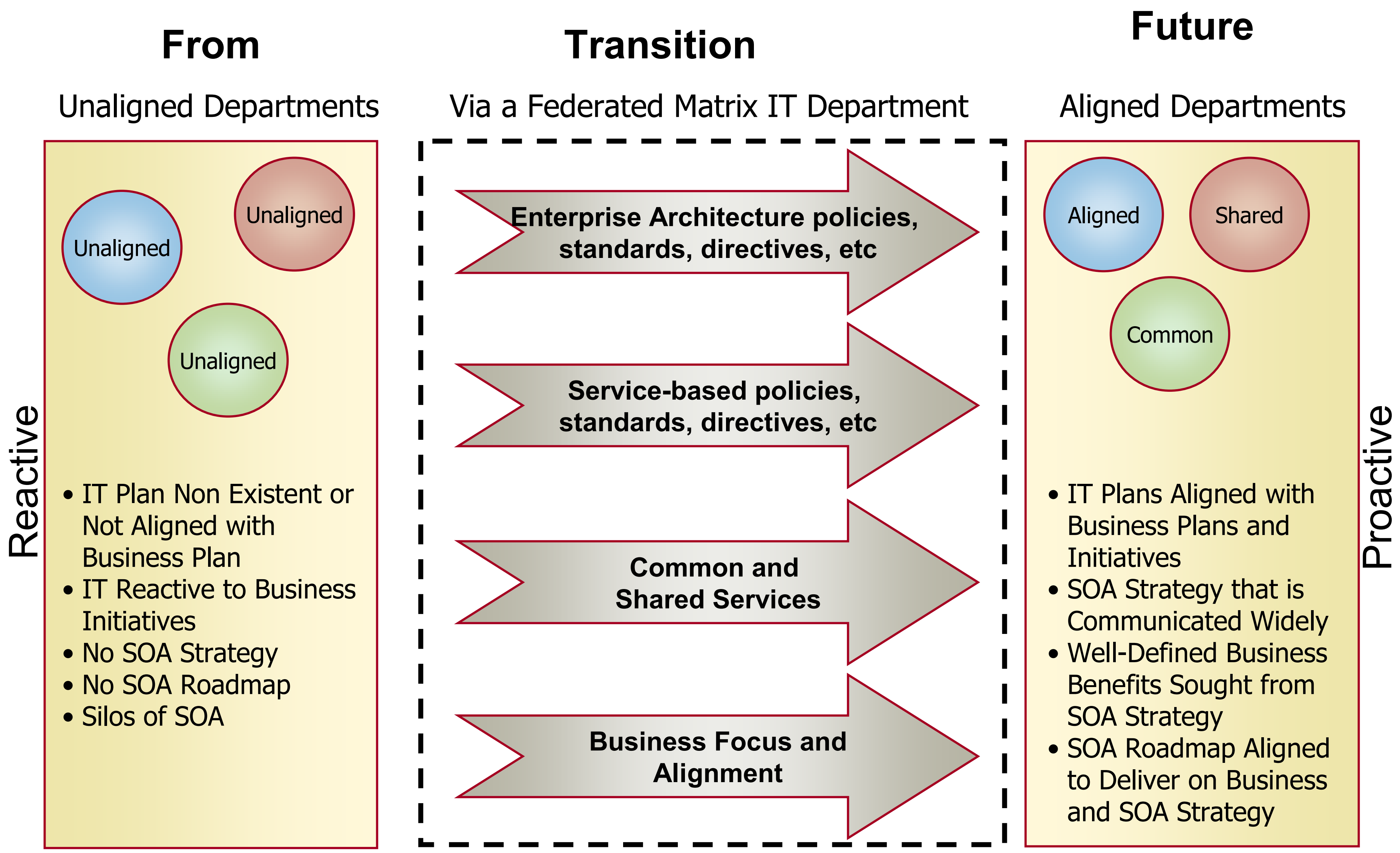
API Development Team

- App´ AP
- App2 AP
- App3 AP
- Avoid Bolt-on Issue

Canonical Model Team

- Establish/Publish Mode
- Version Control
- Based on Industry Standards

Governance: Where do You Want to Go?



From

Unaligned Departments

Unaligned
Unaligned
Unaligned
Reactive

- IT Plan Non Existent or Not Aligned with Business Plan
- IT Reactive to Business Initiatives
- No SOA Strategy
- No SOA Roadmap
- Silos of SOA

Transition

Via a Federated Matrix IT Department

Enterprise Architecture policies, standards, directives, etc
Service-based policies, standards, directives, etc
Common and Shared Services
Business Focus and Alignment

Future

Aligned Departments

Aligned
Shared
Common
Proactive

- IT Plans Aligned with Business Plans and Initiatives
- SOA Strategy that is Communicated Widely
- Well-Defined Business Benefits Sought from SOA Strategy
- SOA Roadmap Aligned to Deliver on Business and SOA Strategy

Operationalizing SOA: Discussion Questions

- How many SOA-based Applications / Services are in Production in your Organization?
- What kind of Capacity Planning measures do you take in your Development of SOA-based applications?
- What types of tools do you use for Performance Measurement? Are they sufficient?
- What is your Funding Approach to SOA? How effective has it been thus far?
- What is your Governance Approach to SOA? How has it worked for you?

SOA UNIVERSE

SOA Knowledge in Action

HURWITZ & ASSOCIATES

liquidhub

Questions and Answers