SOA @ eBay : How is it a hit

Sastry Malladi

Distinguished Architect.

eBay, Inc.

Agenda

- The context : SOA @eBay

- Brief recap of SOA concepts and benefits

- Challenges encountered in large scale SOA deployments

- How is eBay addressing these challenges

- The role of SOA governance

- Summary

The Context : SOA @eBay

- It's a journey !

- History
  - One of the first to expose APIs /Services
  - Support REST style as well as SOAP style
  - Learned a lot from feedback from our customers and operational issues and evolved our technology over time
  - Early adopters of SOA governance automation
  - Continuously improving the architecture with the 3 goals in mind : Business Agility, Innovation and Operational Excellence

- Technology stack
  - Mix of highly optimized home grown core SOA framework as well as best of breed commercial and open source components , integrated together

- Goals
  - Organize the enterprise as a set of reusable business functions
  - Reduce the cost of new features and applications
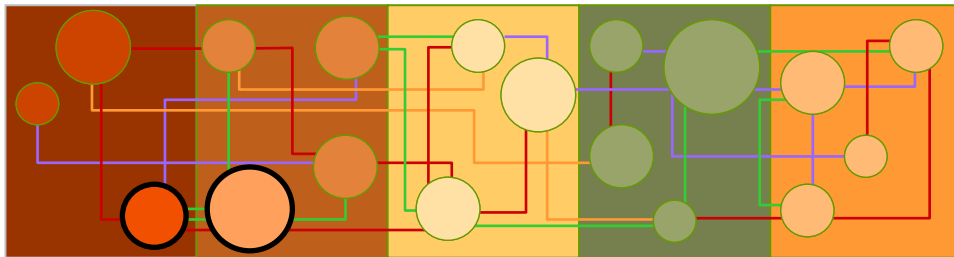 - Encourage and enable new business opportunities

Agenda

- The context : SOA @eBay

- **Brief recap of SOA concepts and benefits**

- Challenges encountered in large scale SOA deployments

- How is eBay addressing these challenges

- SOA governance and its role

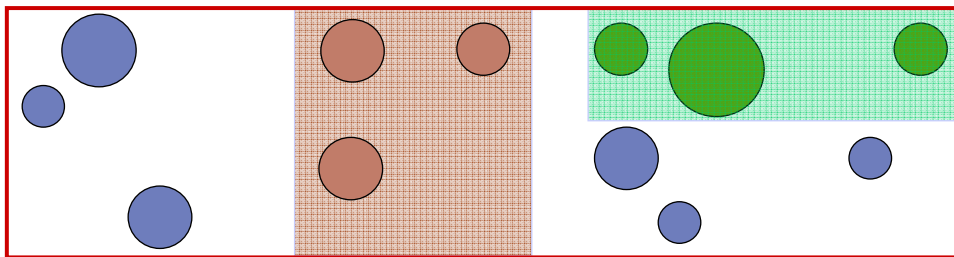- Summary

What is SOA?

SOA is an *Application Architecture evolution*, not a *Technology revolution*

SOA is an architecture to move from brittle, hardwired, application silos that inhibit change…
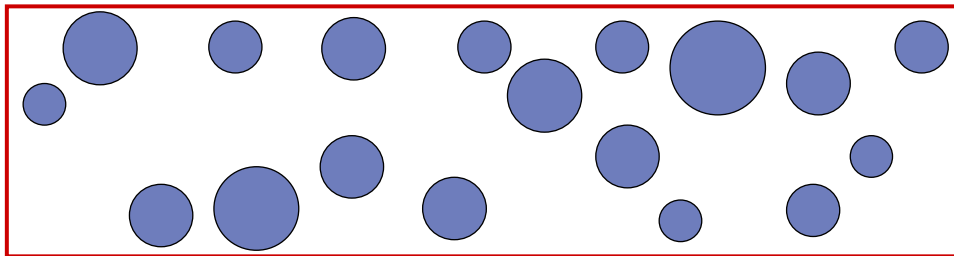
… to shared, reusable, business and application services…

… which eliminates application redundancy and complexity, and enable Business Agility, Innovation and Operational Excellence.

eBaY

SOA - Not just technology

Technology

SOA

Process

- SOA Run-time stack
- Integrated tooling
- Infrastructure services
- Registry/Repository
- Routing

People

- Think SOA
- Promote Reuse
- Biz and IT alignment
- Training & Docs

- Life-cycle and governance Processes for Service Interface Approval, Security, change mgmt, etc.

- PDLC changes

Some common Misconceptions dispelled

SOA is a new technology → It's a new architecture paradigm applied to existing or new technologies

SOA implies Web Services and SOAP → WS/SOAP is not an exclusive protocol. REST style with JSON, NV is equally popular, if not more.

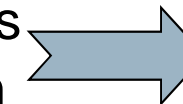SOA is an end in itself → SOA is a means to enable business agility

Services are always developed from ground up → Business always leverage existing functionality, but morph them into services and develop some new services on the way

At service development time, its consumers/usecases are known → Service development may start with some usecases in mind, but they will evolve over a period of time

SOA model from 30,000 feet – You already know !

Decoupled

SOA Framework

Security   Monitoring

Routing/ESB

Service consumer

Data formats

Message protocols

Service provider

Business rules

Policy mgr   Other ..   G11N

WSDL

+ policies

Lookup

publish

Service Registry

optional
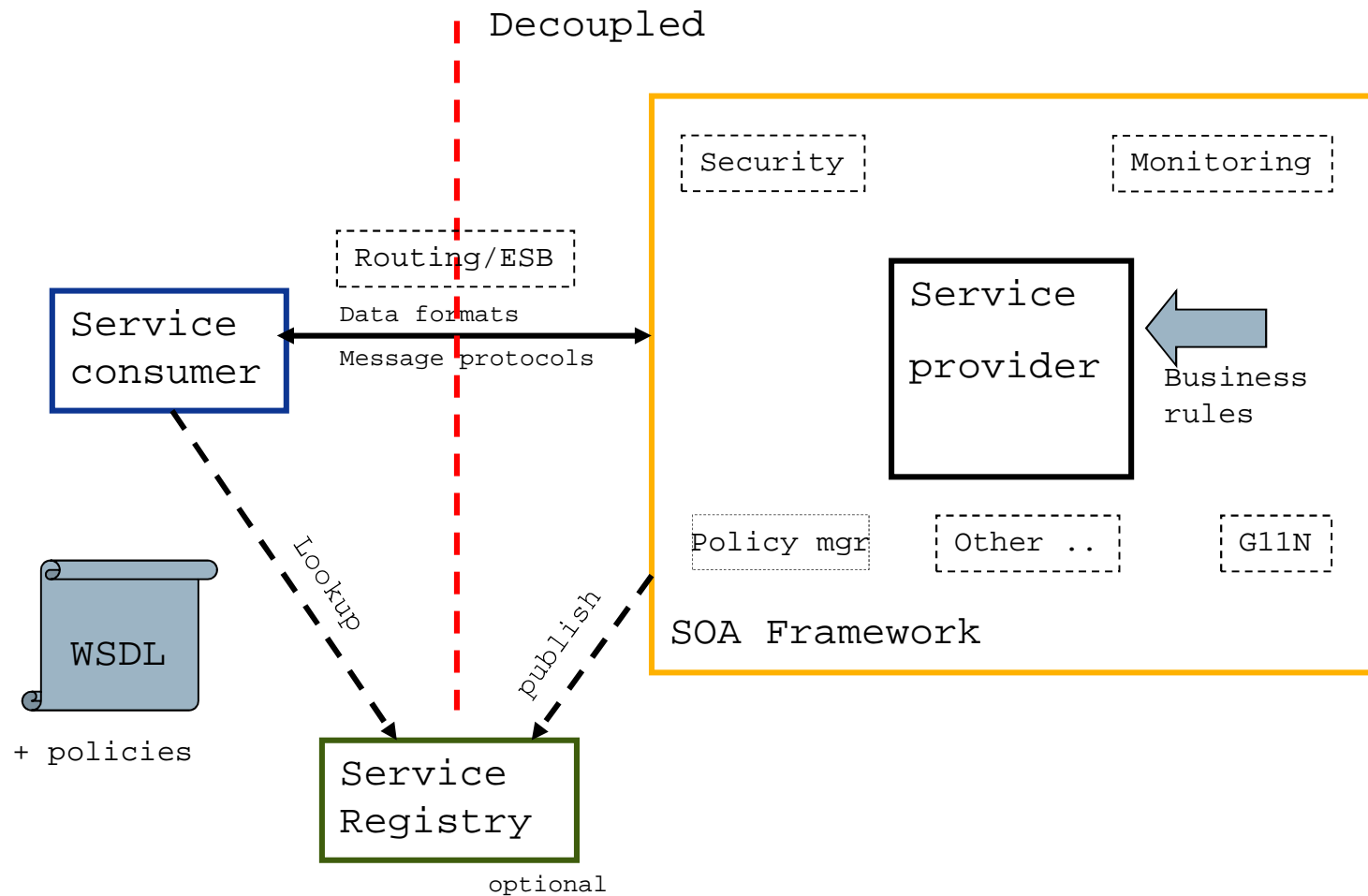
SOA benefits

- Business Agility
  - Faster time to market for new business offerings
  - Quicker response to business change
  - Easier integration with partners and third parties and adjacencies

- Innovation
  - Enable internal and external Innovation

- Operational excellence
  - Reduced cost for any new feature development by service re-use
  - Efficient resource utilization
  - Reduced cost of failure

SOA Benefits – Enabling possibilities

TODAY
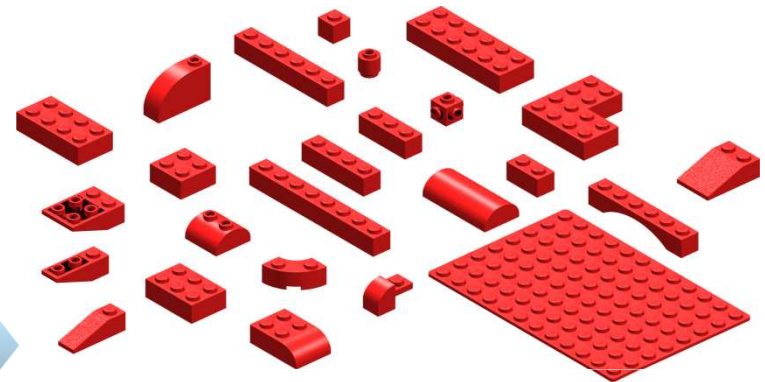
TOMMOROW

FUTURE

Terminology

- Service
  - Well defined piece of functionality, with a defined interface, with fully encapsulated business logic and data, that is managed and exposed consistently, and which is independent of any data formats and message protocols

- Data Binding/Format
  - Also referred to as Wire format
  - The format in which the data is represented on the wire (e.g XML, NV, JSON, BinaryXML, ..)

- Message Protocol
  - The protocol followed by the client and service when exchanging messages (e.g, SOAP). There need not be any message protocol present (especially when communicating with internal services)

- Transport protocol
  - Protocol used at the transport level (e.g HTTP, SMTP)

Agenda

- The context : SOA @eBay

- Brief recap of SOA concepts and benefits

- **Challenges encountered in large scale SOA deployments**

- How is eBay addressing these challenges

- SOA governance and its role

- Summary

General SOA Challenges

- **Technical**
  - Additional latencies due to multi-hop communications
  - Debugging/Tracing is harder
  - Need for efficient Request/Session level caching
  - Increased Security and monitoring challenges
  - Multiple, often competing standards

- **Operational**
  - Developer adoption and learning curve
  - Governance processes
  - Migration of existing Apps
  - Updates to existing tools and processes.
  - Deployment and rollout
  - Measurement of progress and ROI

Further challenges in large scale deployments

- ## Technical
  - Co-existence of old and new technologies during transition phase
  - Supporting internal and external clients that have different protocols/data binding needs, for the same service deployment
  - QoS and SLA management
  - Integration testing
  - High availability and Scalability – high volume and low latency
  - Decomposition of existing apps and migration of legacy services

- ## Operational
  - Version and dependency management
  - Impact to existing operational tools/environment
  - Time to Market pressures
  - Need for a strong yet simple governance process, especially with lots of services and the higher velocity of changes

Agenda

- The context : SOA @eBay

- Brief recap of SOA concepts and benefits

- Challenges encountered in large scale SOA deployments

- **How is eBay addressing these challenges**

- SOA governance and its role

- Summary

Addressing the challenges

- ## Technical

  - Light weight and high performant SOA platform (mix of home grown and commercial/open source components)
  - Unified Testing framework and service virtualization
  - Model-driven Service decomposition
  - Support for REST style as well as SOAP style from the get go (i.e, support for flexibility of protocols/data formats)

- ## Operational

  - Strong and yet flexible governance processes and life cycle management
    - Automated through a service registry/repository
  - Incremental service deployment
  - Strong operational management tools
  - Developer training and incentives for being a good citizen
  - Formal process and tools to measure adoption and progress. Haven't yet formalized the ROI measurement however.
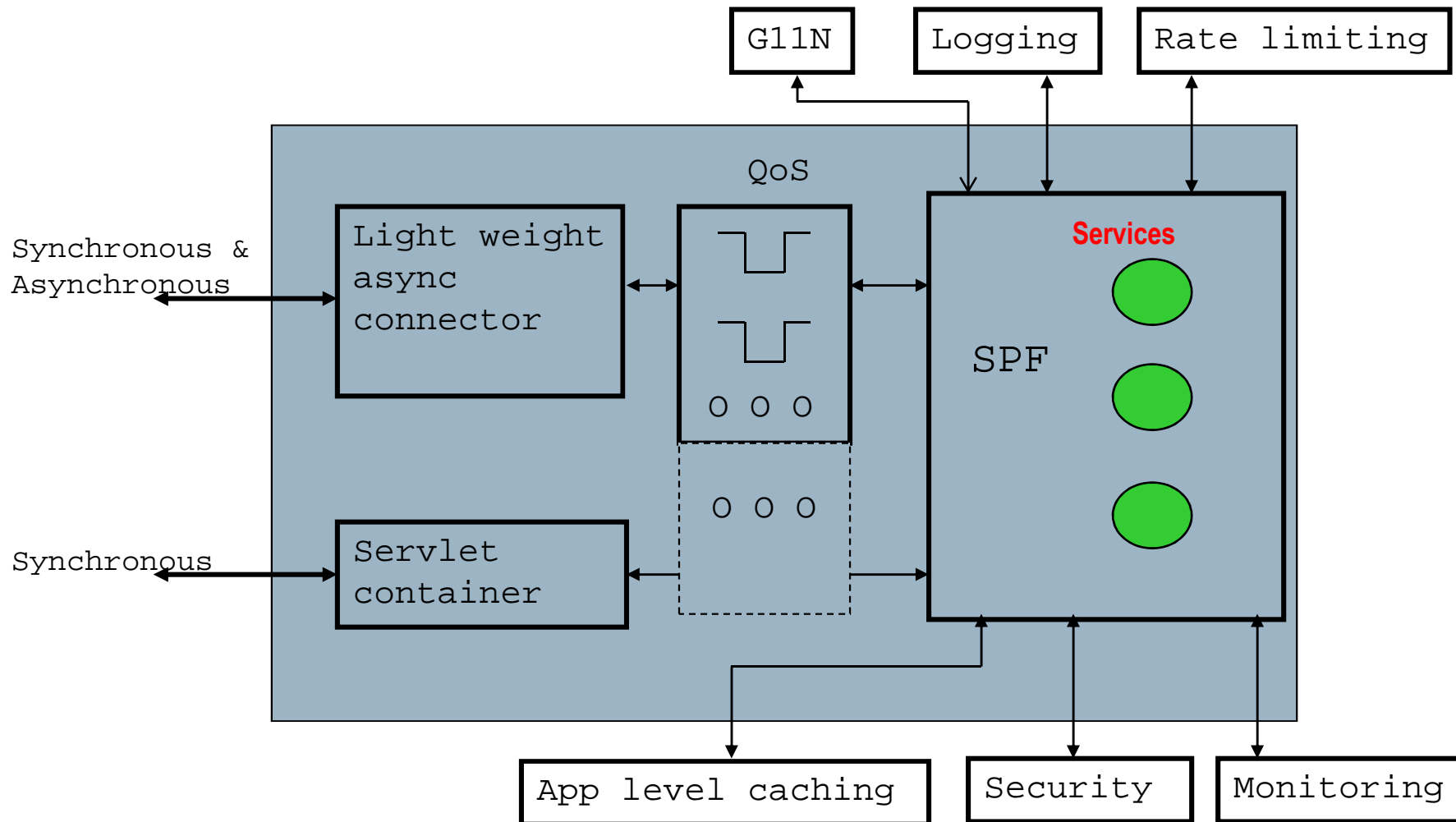
eBay SOA Platform Overview

- A comprehensive SOA Infrastructure platform
  - **Framework** - Highly performant, extensible and light weight framework (overhead < 5 ms)
  - **Monitoring -** A built in monitoring subsystem that is customizable
  - **Security –** XACML/WS-Policy based flexible and extensible authentication and authorization policies
  - **Rate Limiting** - For enforcing capacity, budgeting and traffic control
  - **Service Registry and Repository –** Governance, life cycle management
  - **ESB** - For routing, transformations and mediations
  - **Orchestration engine** – For orchestrating services and other components

- Tooling
  - Developer tools (eclipse plugins) for service/consumer development, types management and error management
  - Operational tools (management, monitoring and alerting)

SOA Framework - Highlights

- A declarative Pipeline based high performance architecture

- Request and Response decoupling

- Protocol and data binding agnostic service
  - The same service instance can be invoked using multiple protocols and data formats
  - No message normalizations or conversations

- Pluggable data formats
  - Out of the box support for SOAP, REST, JSON, Binary XML
  - Streaming and attachment support
  - WSDL with SOAP and HTTP bindings

- Pluggable transports, including local binding

- Declarative , deployment time "Local Binding" option

- G11N, Versioning, Error management, QoS etc.

Model within the App Server

G11N   Logging   Rate limiting

QoS

Synchronous & Asynchronous → Light weight async connector

Services

SPF

Synchronous → Servlet container

App level caching   Security   Monitoring

Pipeline architecture – provider framework

SPF

Logging handler

Global &
Service
Specific
Config files

Auth handler

G11N

Reads

Incoming
request

Framework
servlet

OR

New tomcat
connector

1

Outgoing
response

9

2

8

Server
Message
Processor

MCtx

3

In Pipeline

4

Out pipeline

6

7

Request
Disp

5

ServiceImpl

Resp
disp

Transport

Protocol specific
(e.g SOAP) processors

(de)serialization happens
here if not already done
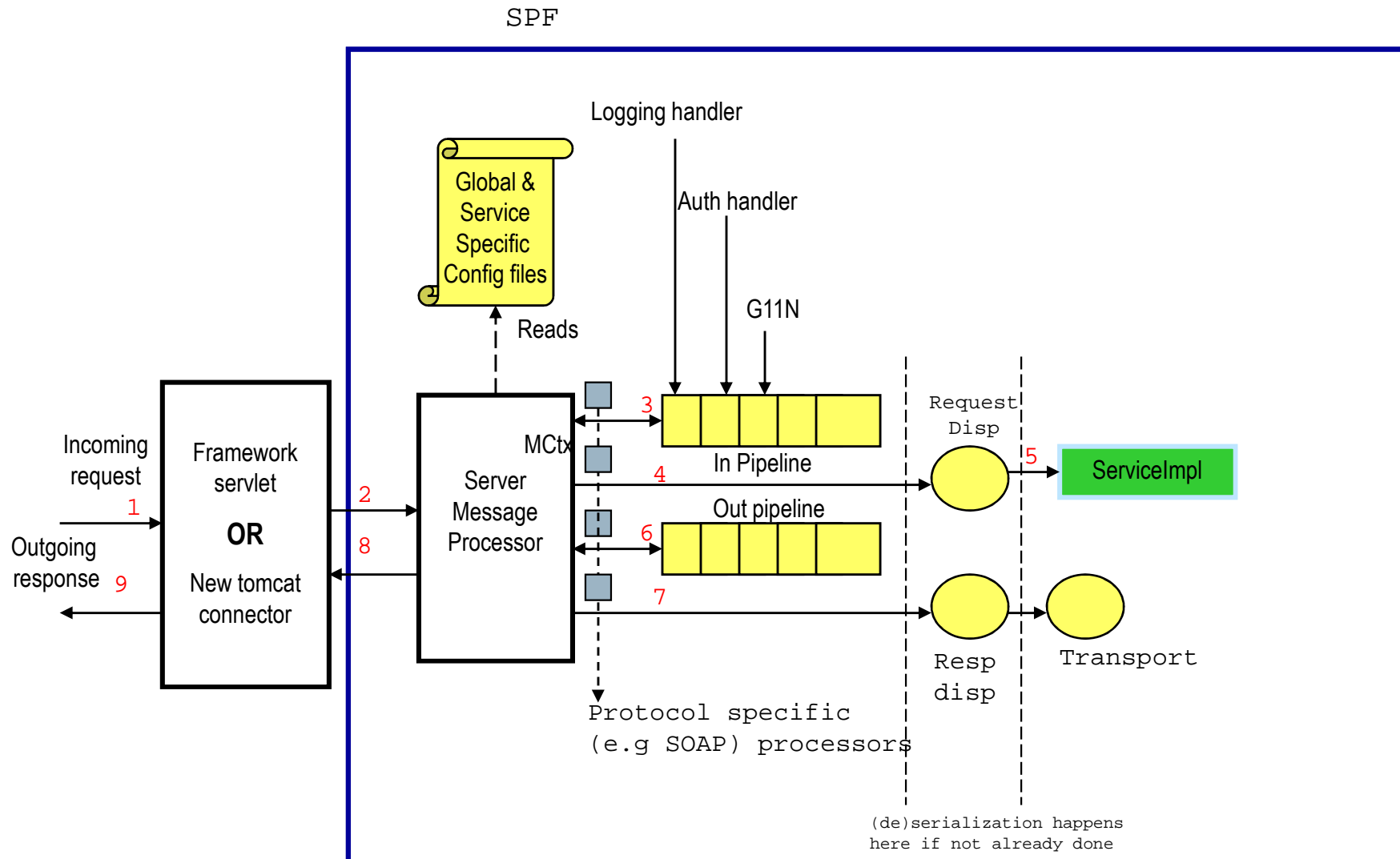
Pluggable data formats using JAXB

Calls from handlers (pipeline)
Or from Req/Resp dispatchers

1

(de)serialize
(incoming)outgoing message

```
(Request/Response) Message

5  Cache (de)serialized
          objects
```

getSerializer/
getDeserializer   **2**
(based on the type)

3

```
(de)serializer factory
```

4

| XML | NV | JSON | Binary XML | Others |

Pluggable (via config)
Uniform JAXB based
(de)serializers

| XML | NV | JSON | Binary XML | Others |

Stax parsers
for each data format

Native and uniform (de)serialization

Native serialization

Uniform interface

Pluggable formats

XML

JSON

NV

....

Other formats

pipeline

Ser/Deser module

XML

NV

JSON

others

Directly deserialized into

Java objects

Passed to

A single Instance of Service Impl

SOA framework

No intermediate format
Avoids extra conversion

ebaY
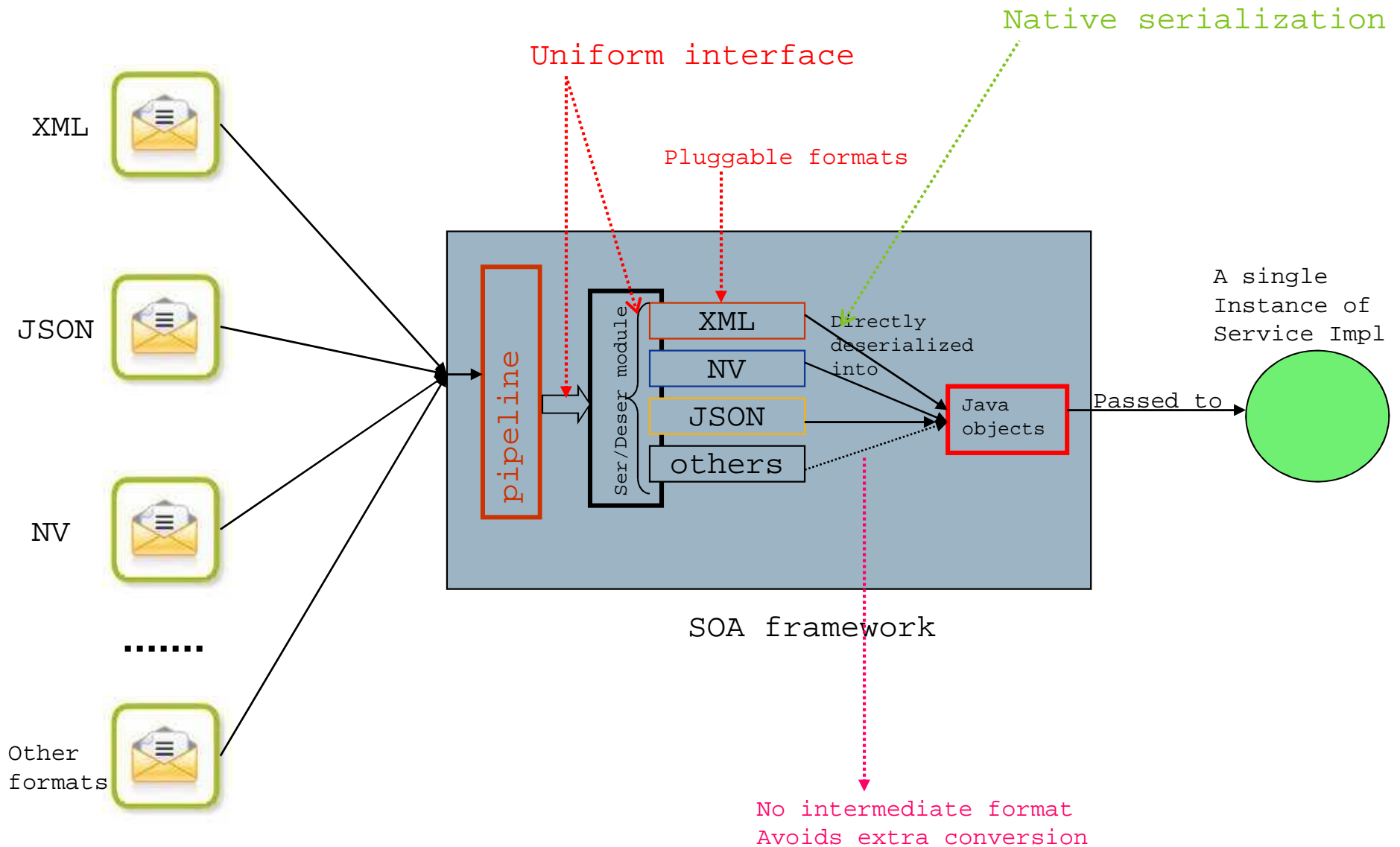
Pipeline Architecture – Consumer Framework

Logging

Auth

G11N

Client Specific Config file

Reads

Transport factory

6

Local transport

HTTP sync and async

SMTP or others

Pluggable transports

3

Out Pipeline

Request Disp

5

Client Message Processor

4

7

In pipeline

8

Resp disp

Client App

SIF API

1

2

10

9

JAX-WS compliant

Protocol specific processors (e.g SOAP)
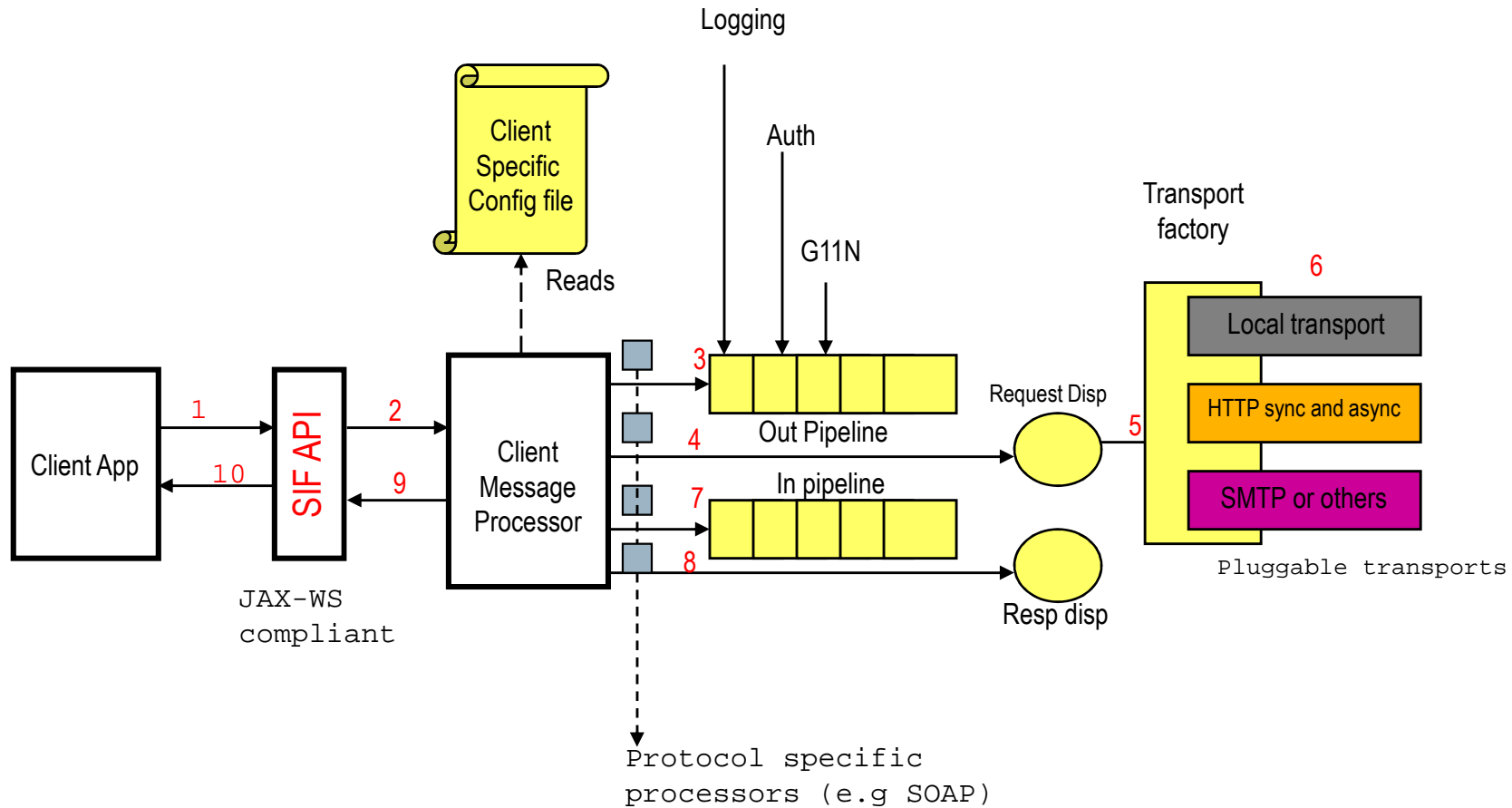
Local Binding
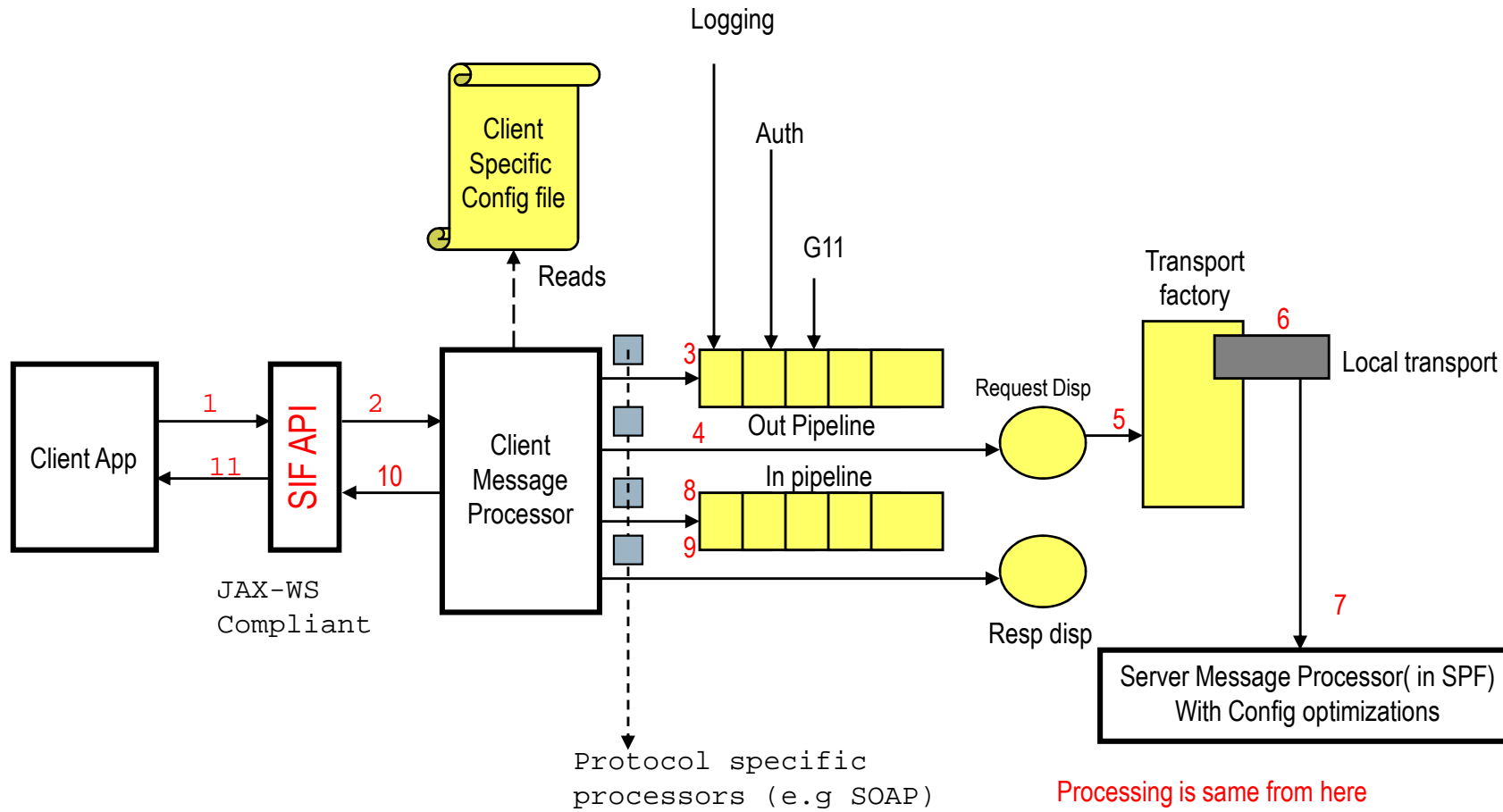
Service Layers and Taxonomy

- Defining a finite set of service layers and their relationships

- Goals
  - Provide a mechanism to build and layer services that have different characteristics
  - Enable service lifecycle management for services with different scopes of impact
  - Clearly establish rules on what functionality can be implemented in which layers
  - Define allowed access patterns between layers
  - Enable better dependency and rollout management

Our specific taxonomy (for example sake only)
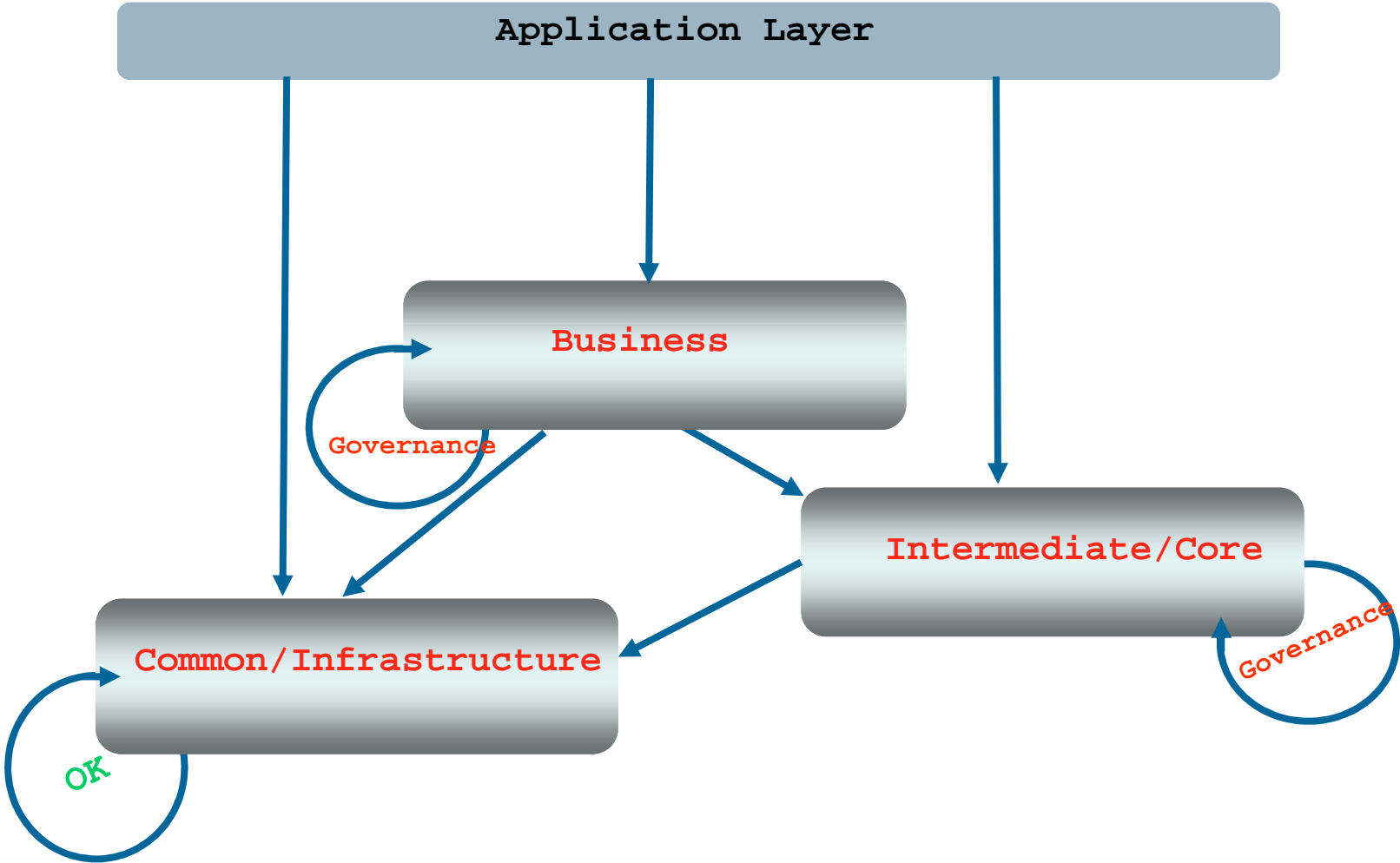
Integrated Tooling

- Developer tooling well integrated with Eclipse IDE
  - Integration with the Service registry and governance processes.
    - Dependencies are checked at design/development time itself
    - Versioning and backward compatibilities are enforced
  - Testing : Generation of tests and a mechanism to enforce successful running of the same

- Deployment tools
  - Dependency checking and rollout in that order
  - Rollback support
  - Local Vs Remote binding
  - Life-cycle management and change notifications

- Operational tooling
  - Management and monitoring dashboard
  - Alerting and SLA management

Note : Some this is still work in progress

Agenda

- The context : SOA @eBay

- Brief recap of SOA concepts and benefits

- Challenges encountered in large scale SOA deployments

- How is eBay addressing these challenges

- **SOA governance and its role**

- Summary

What is SOA governance

- A set of activities related to exercising control over design, development and life-cycle of Service providers and Service consumers, for consistency manageability and for promoting re-use (subset of IT governance)

- Design time
  - Interface and types design review and approval
  - Enforcement of consistency guidelines for contract (interface, policies and behavior)
  - Enforcement of dependency and layering guidelines

- Runtime
  - Deployment policies
  - Security/caching/monitoring/availability policies
  - Reconciliation between what it is (runtime) and what it should be (design)

- Change management
  - Dependency and impact analysis
  - Versioning (both major and minor) of existing services
  - Making sure existing subscribed clients don't break.

- Many industry vendors provide this product
  - Not a very mature space however

SOA governance at eBay

- Service Repository
  - Metadata management and life cycle management of services, data types and Consumers
  - Automated Governance processes
  - Dependency and Impact analysis
  - Closed loop governance

- Service Registry
  - A subset of the repository (without the governance process), light weight access, supports UDDI access
  - Used for routing purposes

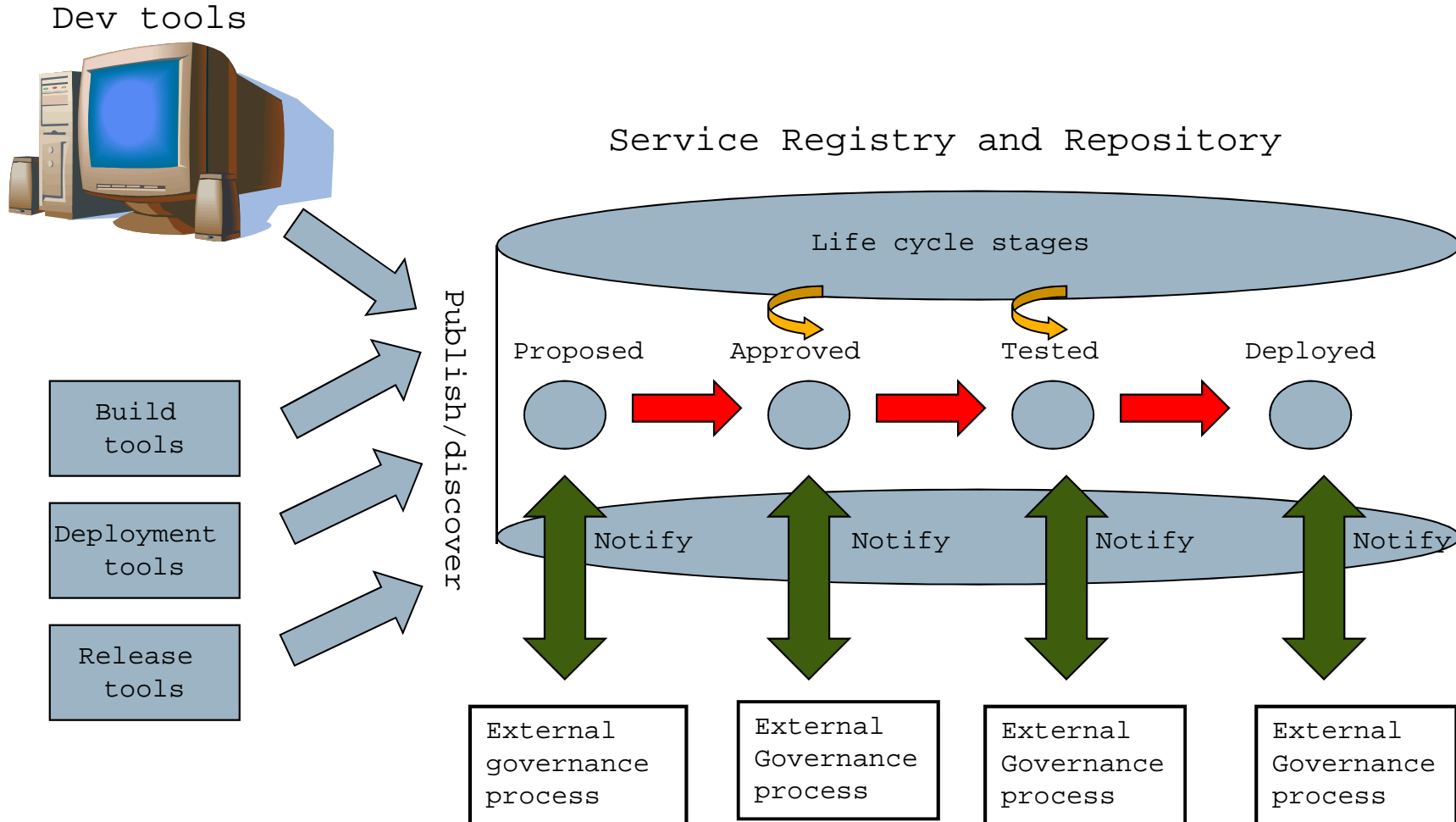- Implementing SOA governance is crucial for success of our large scale deployments

Note : Some of this is still work in progress.

Design time governance

- Defined a light weight, yet flexible process

  - Both services and data types are managed and life cycled

  - An automated workflow logic that that transitions the services through its life cycle states

  - Formal interface (WSDL) and policy (Security, Rate Limiting, Availability) reviews and approvals

  - Automated XQuery based assertions to validate WSDL guidelines

  - Integrated with our SOA eclipse plugins for seamless developer experience

  - Consumer governance to track and manage service consumers

Governance process - Sample

Dev tools

Service Registry and Repository

Life cycle stages

Publish/discover

Build tools

Deployment tools

Release tools

Proposed   Approved   Tested   Deployed

Notify   Notify   Notify   Notify

External governance process

External Governance process

External Governance process

External Governance process

Runtime governance

- Monitor dependencies and policies at runtime

- Feed the runtime data into a reconciliation process that compares what it is Vs what it should be

- Address any discrepancies (closed loop governance)

Agenda

- The context : SOA @eBay

- Brief recap of SOA concepts and benefits

- Challenges encountered in large scale SOA deployments

- How is eBay addressing these challenges

- SOA governance and its role

- **Summary**

Summary

- Moving to SOA model certainly has its benefits, but brings along bigger challenges

- Solving the technology part is relatively easy. Operational aspects are more challenging and need to address them from the beginning

- Up front design and modeling of your contract/interface, including granularity is very important

- Service layering, dependency and version management must be well thought through

- Invest upfront on governance, Testing tools, and developer training

- eBay has been successfully able to apply this strategy and make progress in the SOA adoption journey.