



BPM • SOA • WEB 2.0



SOA Governance is For Life, Not Just a Strategy

Mark Simpson – Consultancy Director, Griffiths Waite

www.griffiths-waite.co.uk

Your Speaker – Mark Simpson



Consultancy Director
Griffiths Waite



- > 18 years Oracle development and architecture experience
- > 1st UK Oracle ACE Director for SOA
- > 1st BPEL project 2005, 1st BAM 2006
- > Regular Speaker
 - UKOUG Conference – 9th Year
 - Butler Group (Ovum) Master classes
 - Gartner, IDC
 - Oracle Open World x 3yrs
 - SOA Symposium
- > Oracle SOA Community
 - Award Winner 2009, 2010 and 2011



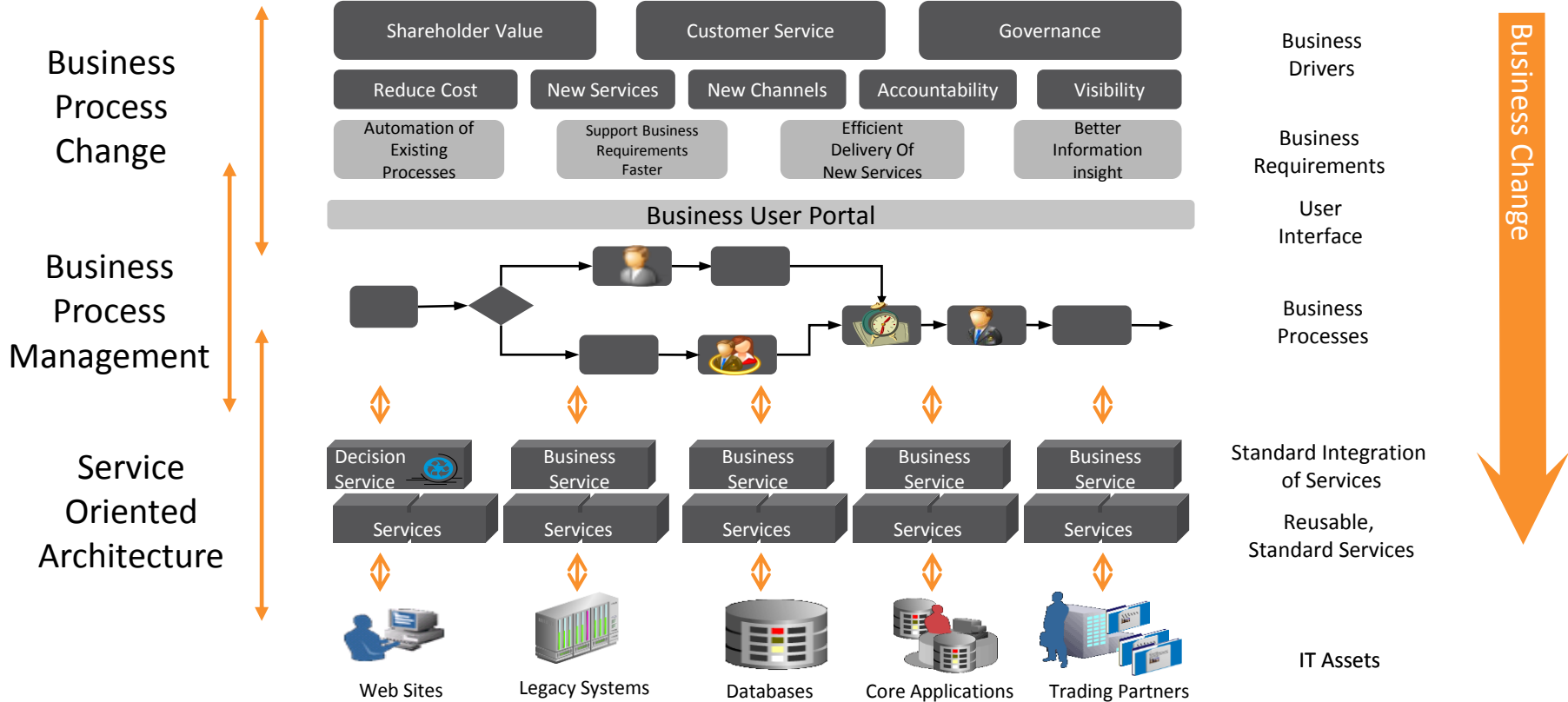
Global Fusion Middleware Innovation Award 2012

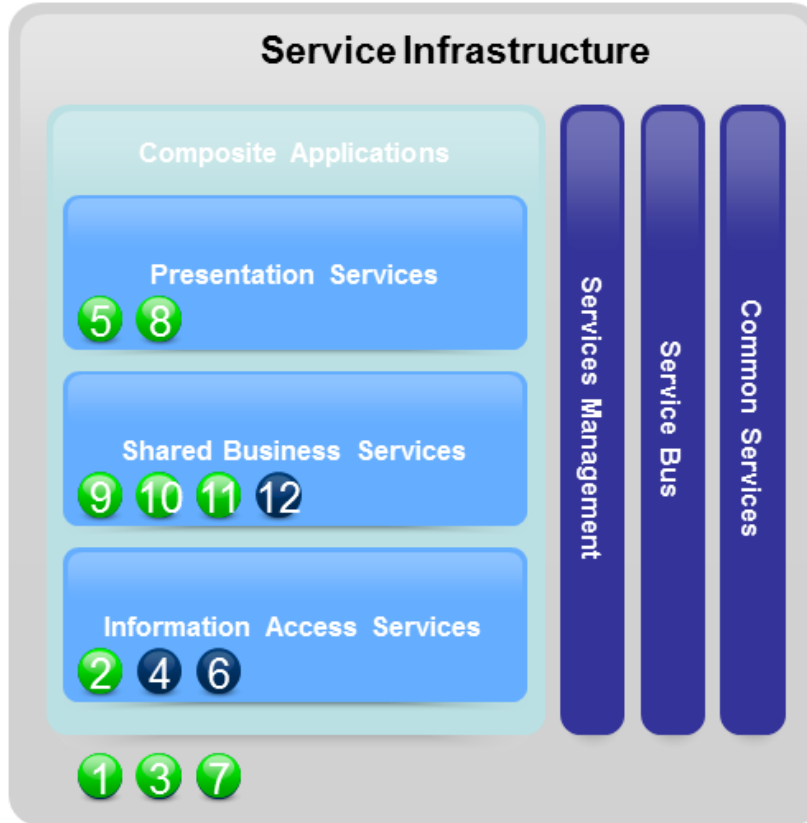
Agenda

- > The promise of SOA
- > Definition of Governance
- > Why govern SOA
- > Start governing now
 - 12 point action plan
- > Key Takeaways



The principle of SOA – Align with Business Model





- Unshared
- Shared

SOA Governance

SOA Governance is an agile, efficient **decision and accountability framework** to effectively direct and assist in realizing the **benefits** of SOA, while encouraging a certain **cultural evolution** in how an organization delivers IT to the enterprise.

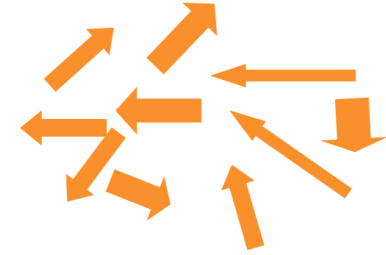
> Definitions

- To ensure only one way to carry out business function
- To promote and enforce **adherence to SOA Reference Architecture**
- Providing a set of principles to ensure there is **no disjoin between the Business Model and the IT Service Architecture** to support this
- Governance as an enabler **accelerates roll out** of SOA
- Ensuring and **validating** that assets and artefacts within the architecture are operating as expected and maintaining a certain **level of quality**.
- Providing policies to **influence** behaviour towards **alignment with business objectives**
- Oversight of SOA assets covering **People, Process and Technology**

Providing Structure / Direction to a SOA Ambition

> We are planning on doing SOA but ...

- What do I need more than web services?
- We don't know how to define or identify a SOA service
- We don't know if our investment in EAI helps us
- Which elements of SOA infrastructure do we need first?
- We have lots of examples of SOA going on ... all different
- How do we do service versioning?
- How does SOA relate to Enterprise Architecture?
- How do we link systems in applications into SOA?



***These are all technical aspects
of a SOA IT strategy which
can be answered in a
SOA Reference
Architecture and Governance Strategy***



SOA Governance Why is it needed? Without It ..

- > Service anarchy - Poor services, low-reuse, wrong services, perpetuation of silos
- > Business Solutions fail to use Services - No promotion, no change in project delivery, no trust or confidence in services
- > High cost of operations - Duplicated infrastructure, no standardisation
- > No measured benefit - No consistency “everyone’s doing their version of SOA”

Wild West SOAs	Duplicated SOAs	Shelfware SOAs
<ul style="list-style-type: none">• Services proliferate wildly• No formal service-definition process• Nobody knows how many services are in place, where they are or what they do• No leverage and no reuse. <p>Extremely difficult to fix and gain control over</p>	<ul style="list-style-type: none">• Things seem to work well• Many services have been duplicated twice or more• Little reuse• High maintenance costs <p>Companies are often reasonably happy with these SOAs, even though savings would multiply if duplication reduced</p>	<ul style="list-style-type: none">• SOA is implemented• Few applications actually use the public services• Point-to-point, unstructured integration• Little buy-in from several business units, <p>SOA is a waste of resources in this context and won't deliver benefits</p>

- > Projects take Precedence over Sustainability
 - Short term opportunistic project view
 - PMO priorities with no architecture viewpoint
- > Business Projects don't leverage services
 - No visibility
 - Services not quite fitting requirements
 - Trust of Developed Services
- > Timing
 - We will start managing the assets after this project
- > Misalignment of Projects
 - Service not ready and tested to be reused when required
 - Need functionality now, don't want to add external risks to the project
- > "SOA is just web services"
 - Proliferation of low level, application specific services
- > Funding
 - We have already invested in SOA (technology), just deliver projects

SOA Governance throughout

Planning Governance

- Ownership / Roles
- Reference Architecture
- Sharing Model
- Business Alignment
- Capability Mapping

Design Governance

- Consistency
- Reuse facilitating
- Design Guidelines
- Impact Analysis
- Service Lifecycle

Runtime Governance

- Discovery
- Policy Enforcement
- Versioning
- Usage, SLA and Performance
- Service Monitoring and Management

The Clock is Ticking.. Get Control of your SOA now

12 Measure to promote success

11 Optimise and Manage

1 Understand your SOA Goals

10 Automate !!

2 Define a Service

9 Build For Change

3 Build Ref Arch

8 Know your Services

4 Development Process

7 Service Testing

5 Define Service Roadmap

6 Define Service Lifecycle



1 Understand your goals

> Increasing Reuse of Assets/Services

- New applications can be built in a way that not only reduces the cost of development but also maintenance over time.

> Reducing Integration Expense

- Loosely coupled Services can handle a wider range of interactions in a more flexible manner than API-based integration.

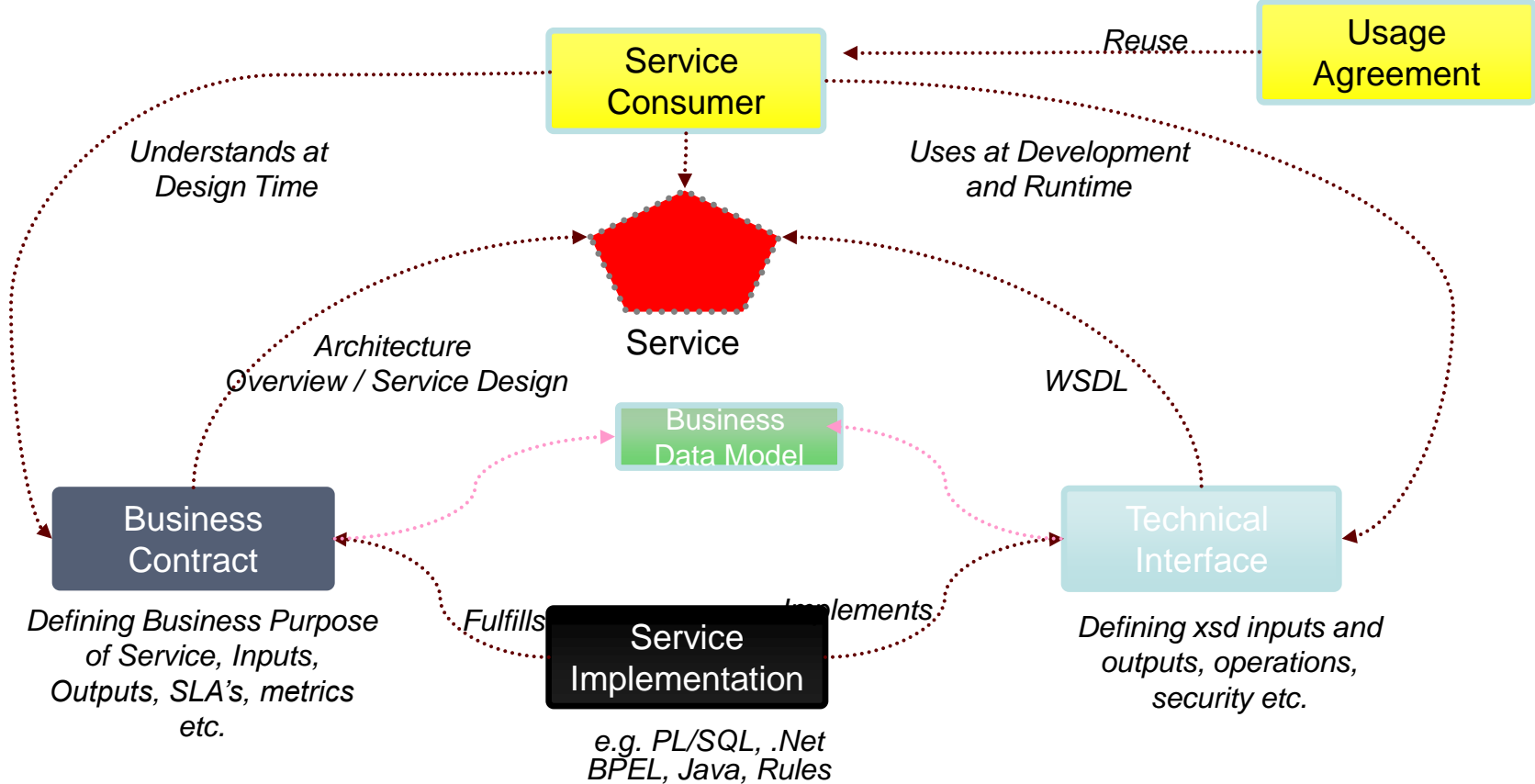
> Increasing Business Agility

- Technology centric benefits such as simplified integration and improved reuse provide business with greater flexibility.

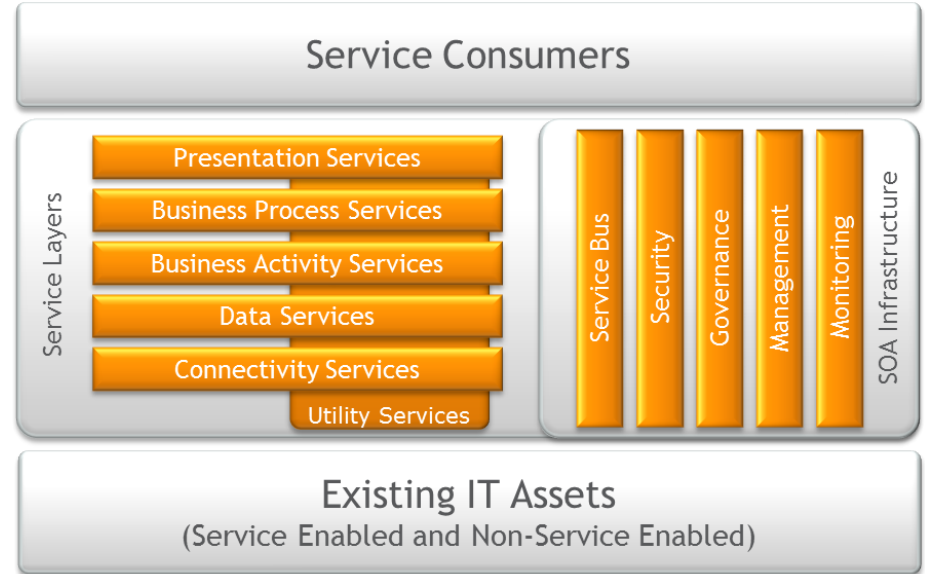
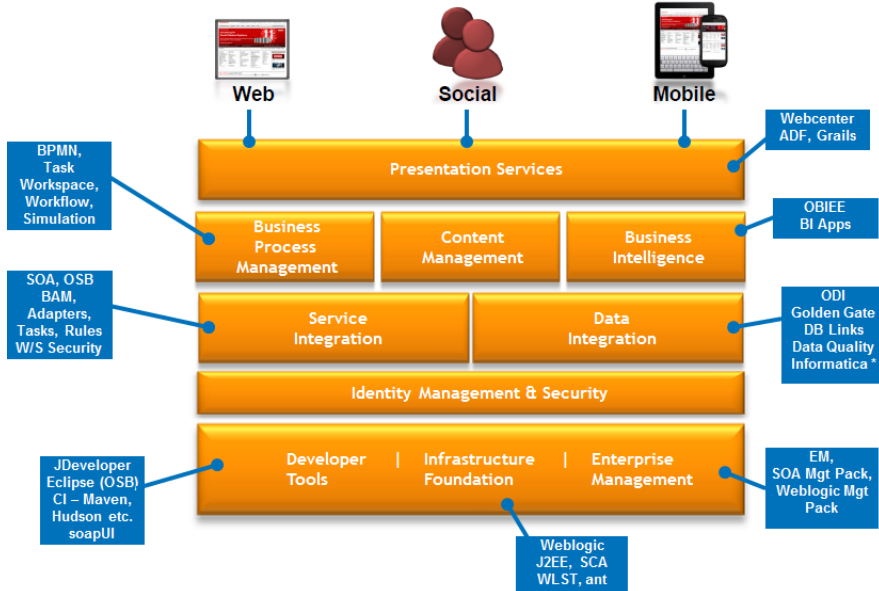
> Reducing the Cost of Change

- Change is isolated to certain services or configuration of service reducing the breadth of impact of business driven change

2 Define a Service

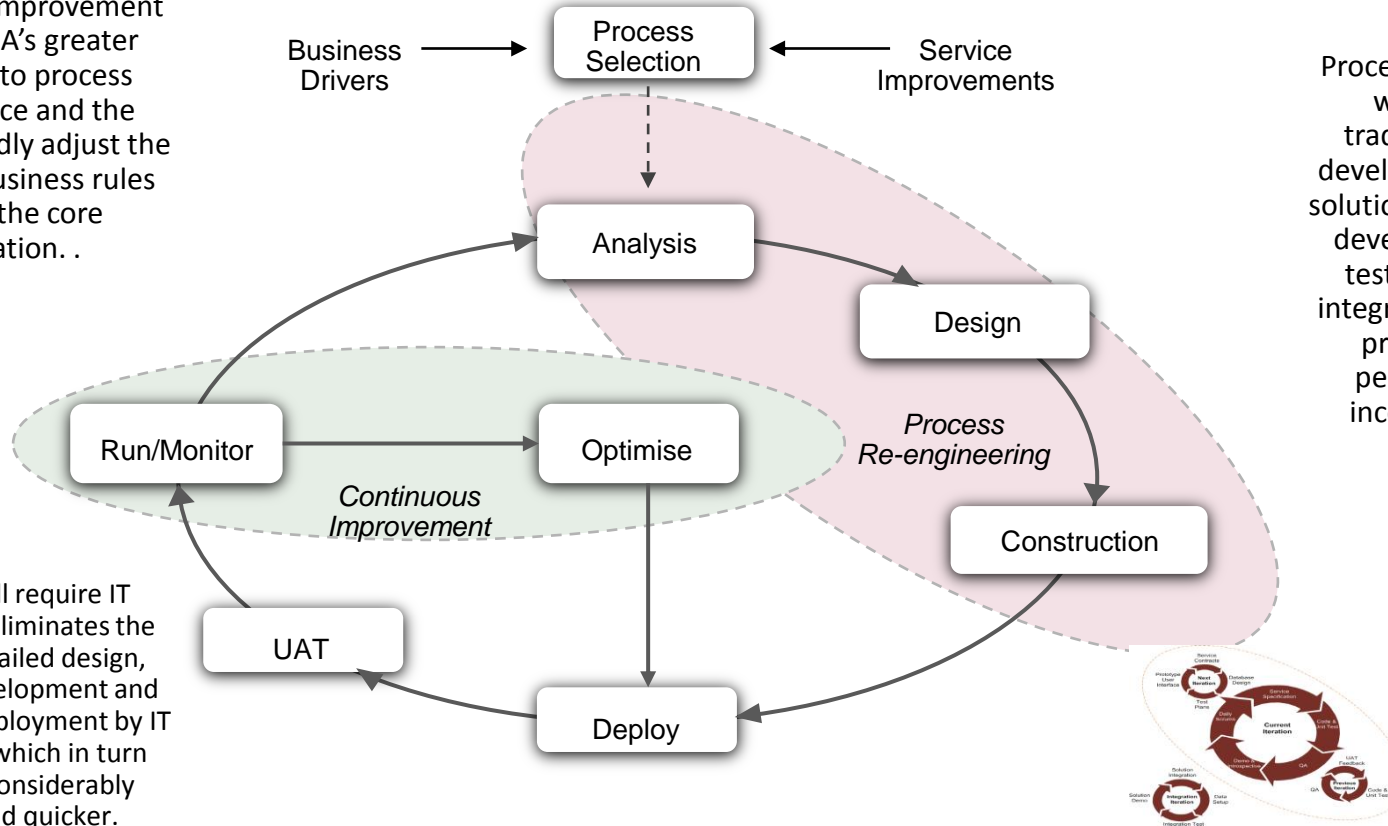


3 Build a Reference Architecture



4 Implementation Methodology

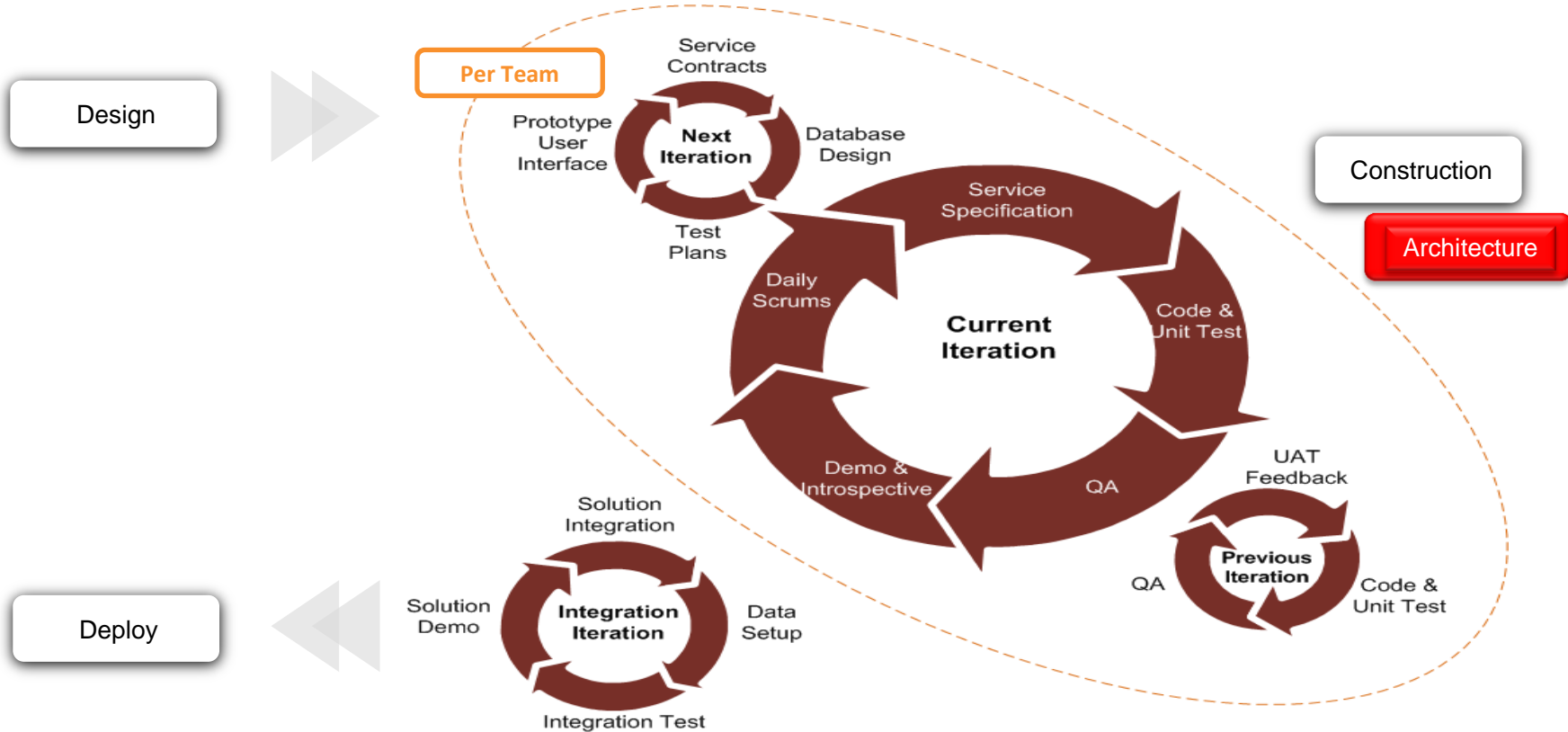
Continuous Improvement exploits SOA's greater visibility into process performance and the ability to rapidly adjust the process / business rules outside the core application. .



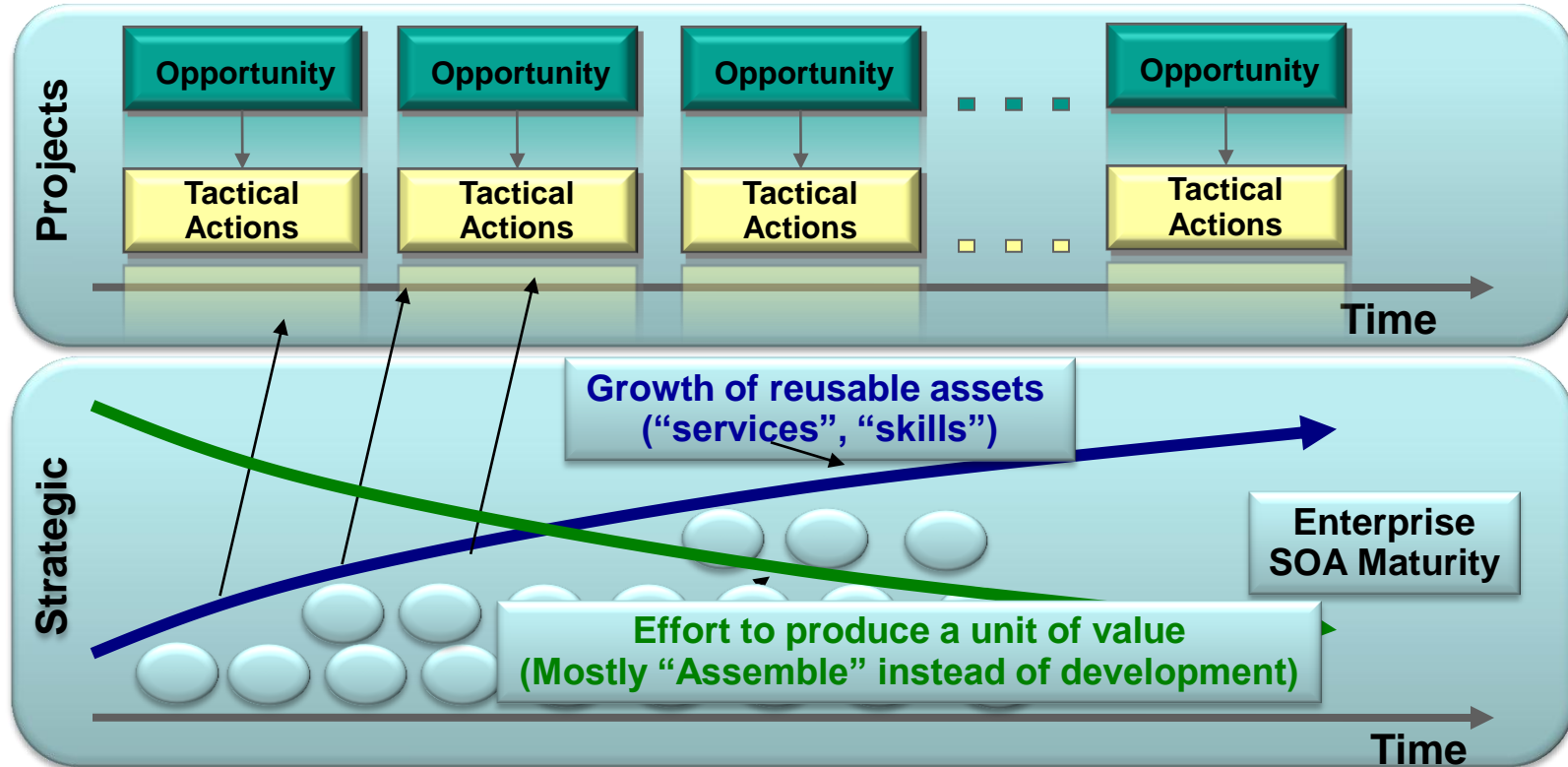
Process Re-engineering will encompass traditional software development – detailed solution design, software development, system testing and systems integration – but from a process /services perspective whilst incorporating Agile principles.

This will still require IT support but eliminates the need for detailed design, software development and QA prior to deployment by IT operations, which in turn should be considerably simpler and quicker.

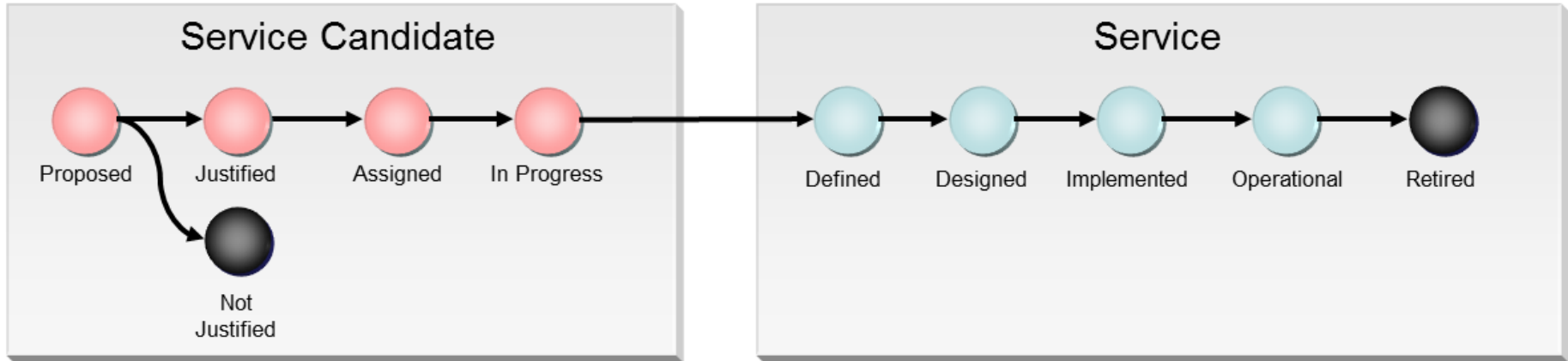
4 Construction and Architecture



5 Consider Portfolio of Services, not just Projects

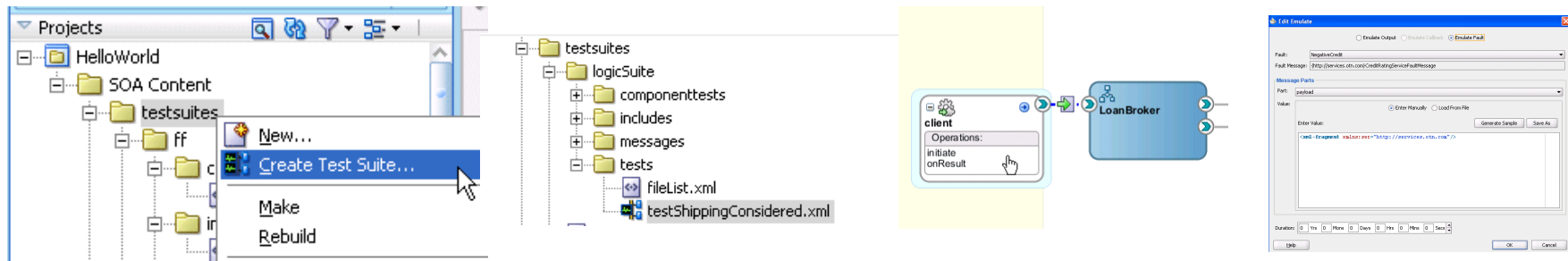


6 Service Lifecycle Stages



7 Service Testing

- > Write Service Acceptance Tests
- > Automate In/Outs, key process paths through SCA Unit Tests
- > Understand Assertions, Emulations, Waits, Fast Forward
- > Add business scenario tests – e.g. soapUI Test Suites, CAVS, OATS
- > Document Test Scripts with the design documentation – link from Asset repos.



“ I won't reuse a service unless I can test it first”

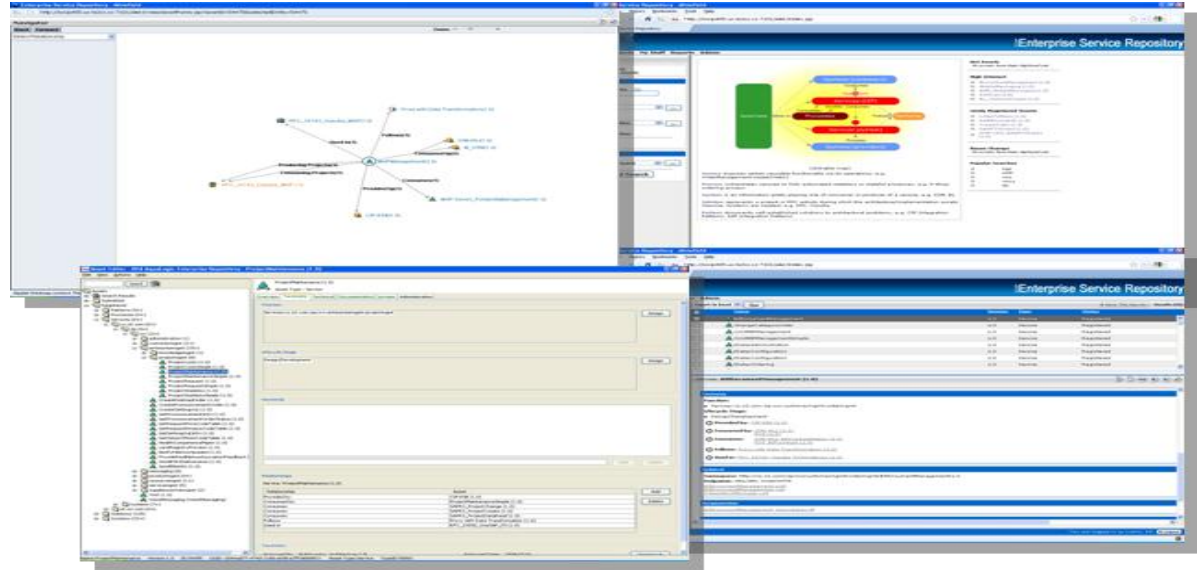
8 Know your Services – Repository

> Functions

- Manage service assets and artifacts: service publication & service discovery
- Apply governance policies, tasks and roles.
- Trace asset change history and provide version control.
- Link tools in the service lifecycle via the repository

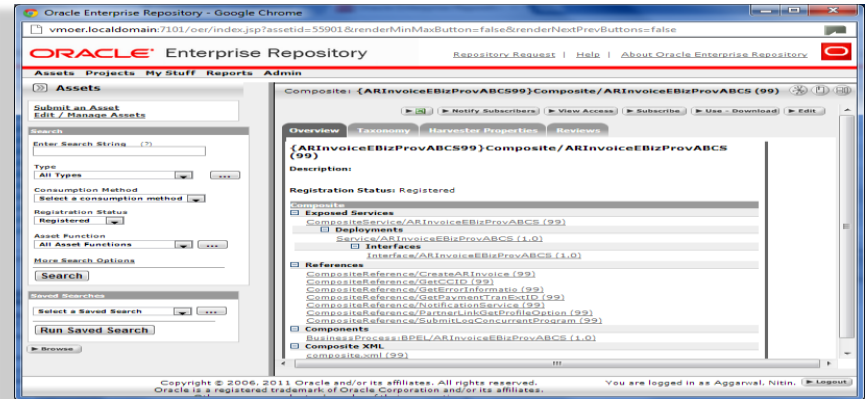
> Automation, rules, policies, and incentives... ...to get a desired behavior

- People
- Process
- Technology

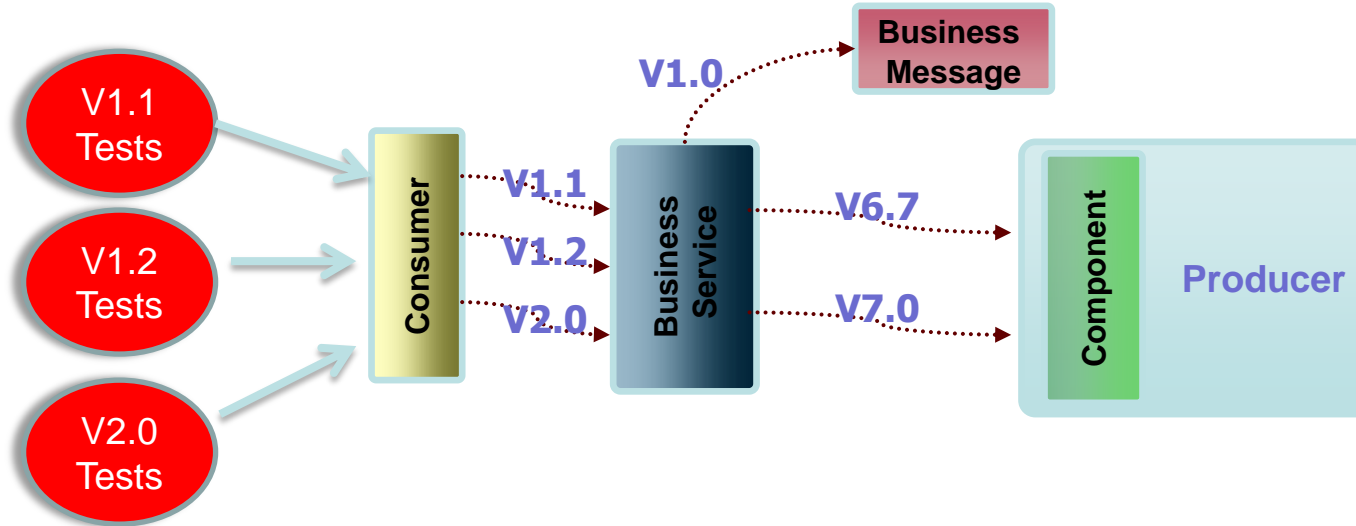


8 Oracle Enterprise Repository

- > Register and link your SOA assets (harvest)
- > Map your service lifecycle / design workflow
- > Library of assets for design / development
- > Apply policies for good design practice
- > Perform impact analysis of pending change
- > Link to runtime metrics
- > Apply taxonomies to tag and search for assets



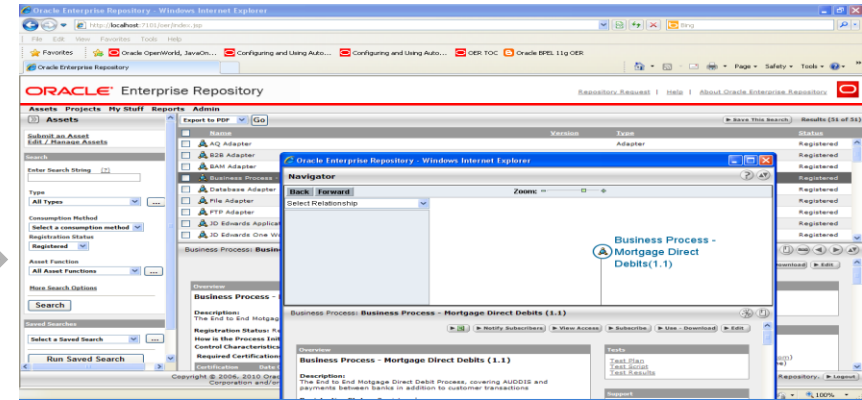
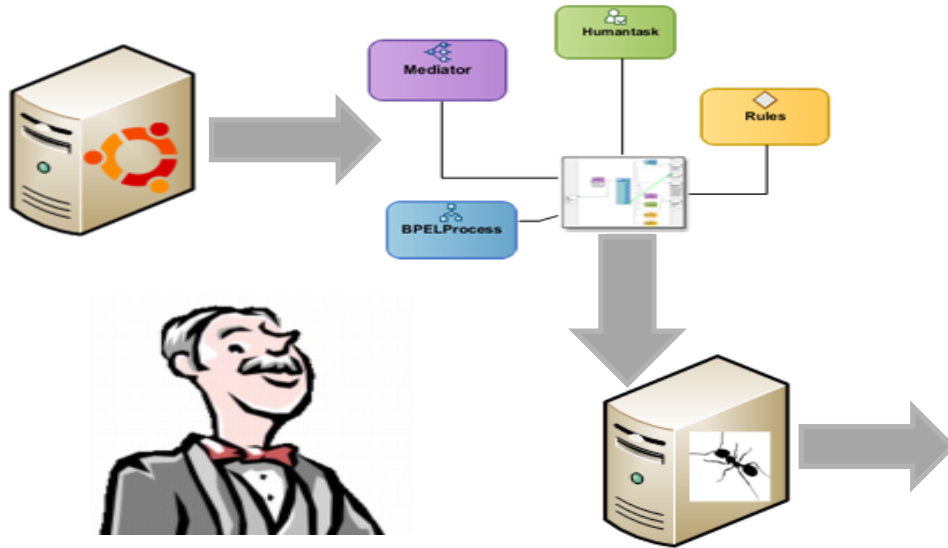
9 Build For Reuse and Change



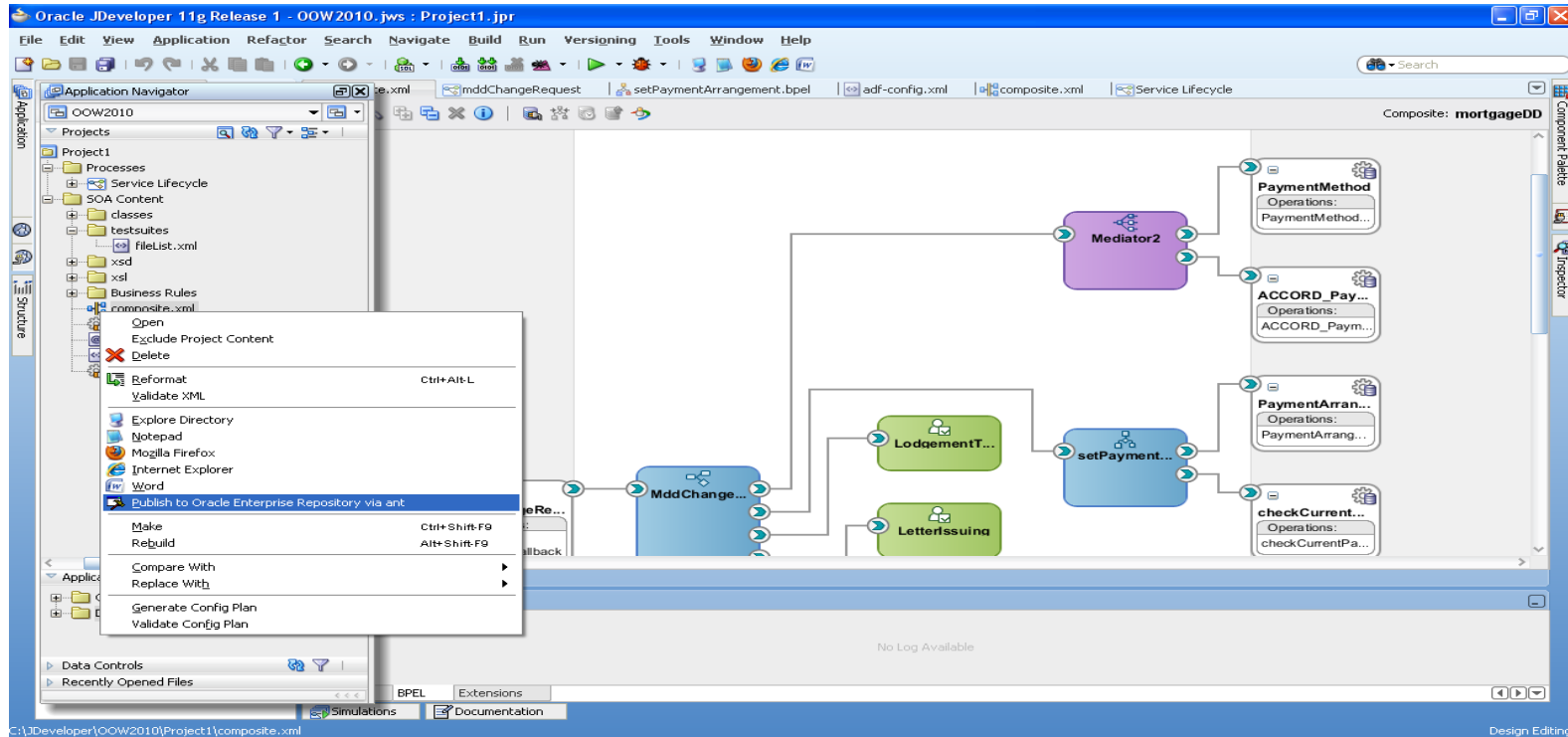
Automated Tests become first class assets to support Service Change

10 Automate the deployment process

- > Include OER and SCA tests in your build scripts and CI
- > Use OER policies as checkpoints for shared services



10 Automate publishing to OER



Oracle JDeveloper 11g Release 1 - OOW2010.jws : Project1.jpr

File Edit View Application Refactor Search Navigate Build Run Versigning Tools Window Help

Application Navigator: OOW2010, Project1, Processes, Service Lifecycle, SOA Content, classes, testsuites, fileList.xml, xsd, Business Rules, composite.xml

Composite: mortgageDD

Component Palette: Inspector

Context Menu:

- Open
- Exclude Project Content
- Delete
- Reformat (Ctrl+Alt-L)
- Validate XML
- Explore Directory
- Notepad
- Mozilla Firefox
- Internet Explorer
- Word
- Publish to Oracle Enterprise Repository via ant**
- Make (Ctrl+Shift-F9)
- Rebuild (Alt+Shift-F9)
- Compare With
- Replace With
- Generate Config Plan
- Validate Config Plan

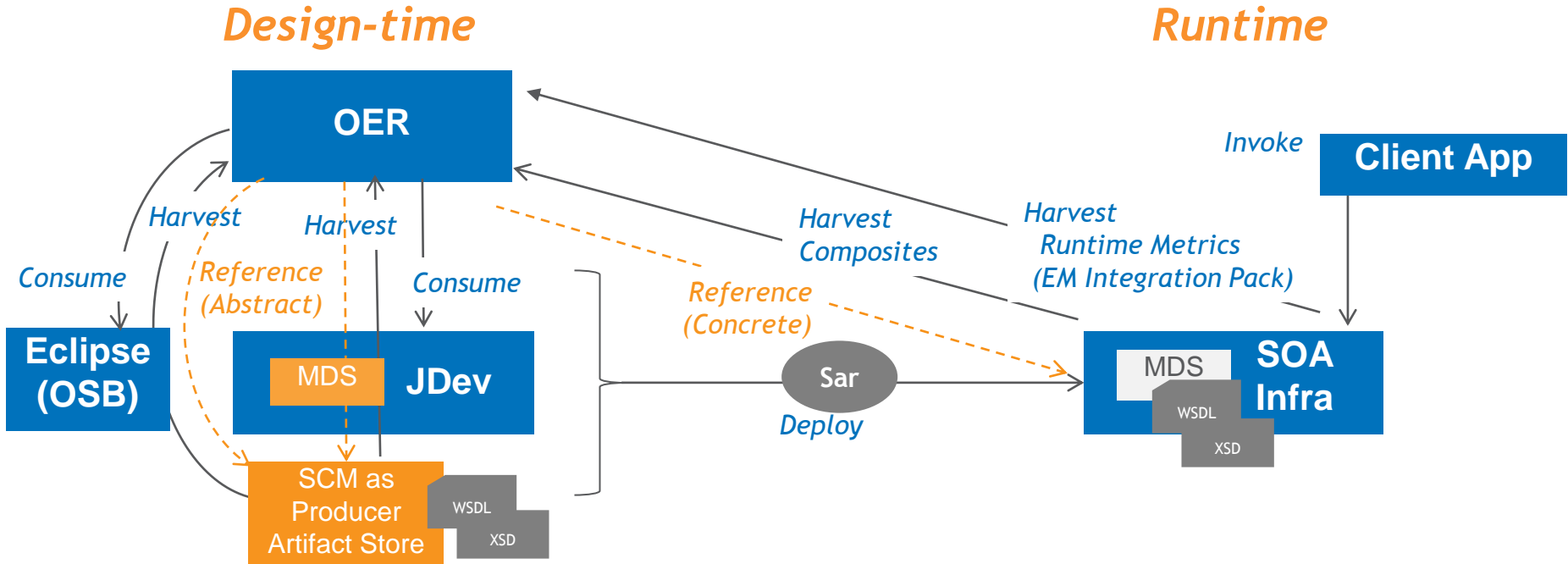
BPEL Extensions

Simulations Documentation

C:\JDeveloper\OOW2010\Project1\composite.xml

Design Editing

10 Automate Design and Runtime Harvesting

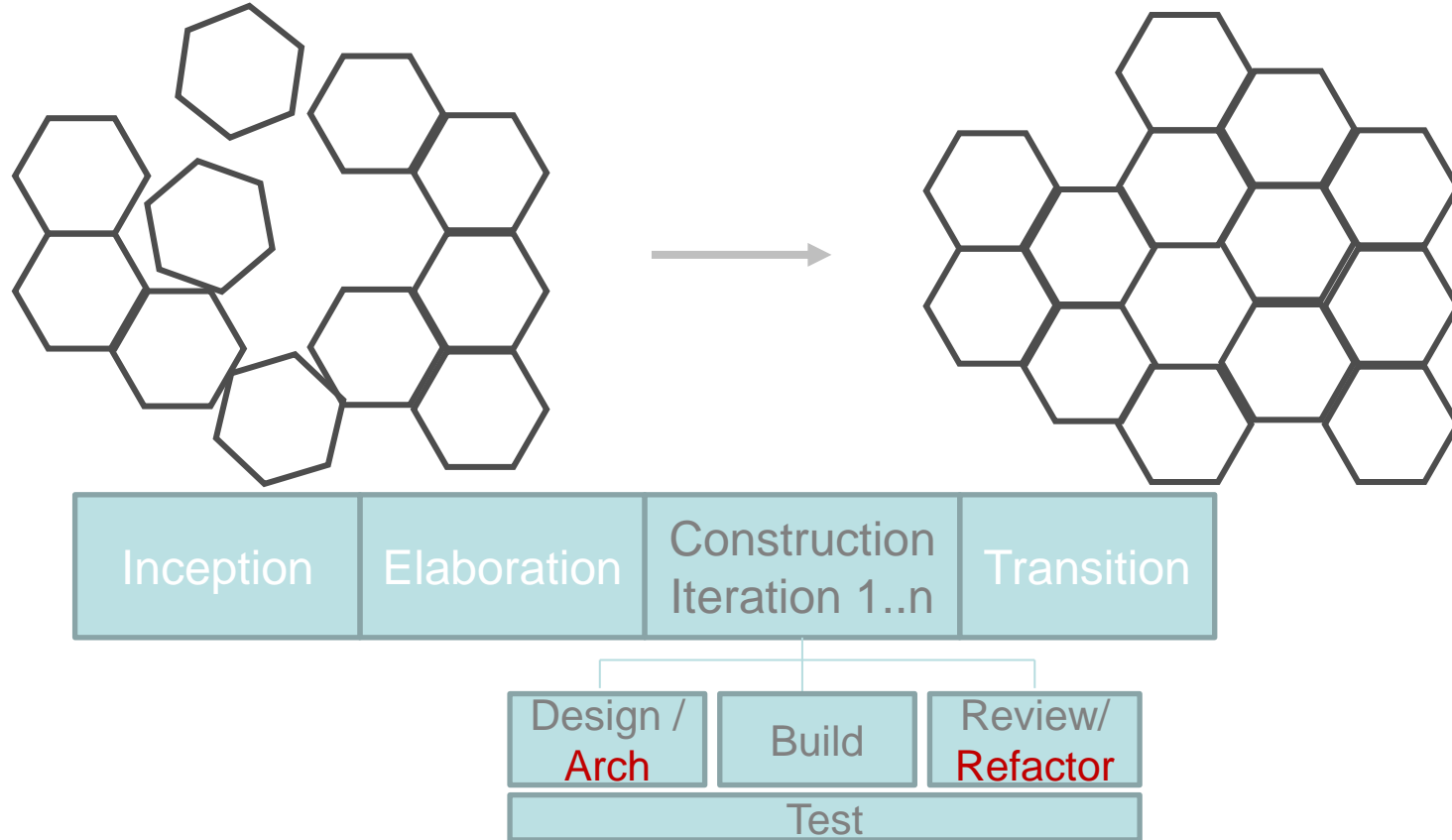


SOA is a fundamental change ..

- > From dedicated to shared resources
 - > Ownership changes, goals change
- > Design for shared use
 - > The need for iterative development
- > From monolithic to composite
 - > But reuse is not the only measure of success, align services with business
- > From static to dynamic
 - > Design for constant change, Autonomy and impact analysis

Govern and Refactor to re-align with SOA principles

11 Realign Service with Architecture



11 SOA Runtime Management

- > **Managing multi-tier transaction flow**
 - Span shared components / services
 - Deployed across several tiers in different containers
- > **Performance and visibility into SOA services**
 - Business monitoring of messages
 - Flow Trace, audits, error reporting
 - Usage Statistics
- > **SOA Environment Control**
 - Availability of Services
 - Access to Services
- > **Administrative tasks**
 - Cloning , smoke testing
 - Automating time consuming and error prone tasks

Risks

- Business Downtime
- Higher Maintenance Costs
- Less Agility



12 Measure and Promote Success

- > Focus on more than just reuse metrics, although these will help you feed productivity measures
- > Define your success indicators up front, ensure these feed into PMO prioritisation
 - Monetary Savings
 - Architectural Debt Reduction
 - Customer Value
 - Supporting Change
- > Monitor Business Performance against different releases of services.
- > Keep a map of delivered “Business Capabilities”

Key Takeaways

- > Apply the level of governance that suits your SOA Strategy.
- > Establish a Reference Architecture, Governance Policy, Test Automation and Versioning Strategy
- > Change brings complexity with lots of moving parts – and enabling change is the prime motivation for SOA
- > Ownership issues must be resolved, they have always been there just now they are more visible
- > Governance is not achieved by technologies alone – in fact it is more important to get the organisation and process established
- > **Whenever you get started on SOA governance – you will end up wishing you had started earlier!**



Twitter:
mark_gw

Email:
mark.simpson@
griffiths-waite.co.uk

Wednesday

13:55 - 14:40

**SOA Governance
is For Life. Not
Just a Strategy**

Mark Simpson,
Oracle ACE
Director - Griffiths
Waite



Need to know more?



Service Oriented
Architecture 24/7

How far is your business from implementing SOA?



SOA247.COM

WEB20247.COM



Web 2.0
24/7

Next Generation User Experience.



Business Process
Management 24/7

Get your highest value business assets working.



BPM247.COM