IBM SOA Technology Summit

Moving Ahead With SOA

Business services development

Francis Geysermans, Architecte, e-business Solutions Center, IBM

SOA on your terms and our expertise

ON DEMAND BUSINESS™

The IBM Vision for Business Driven Development

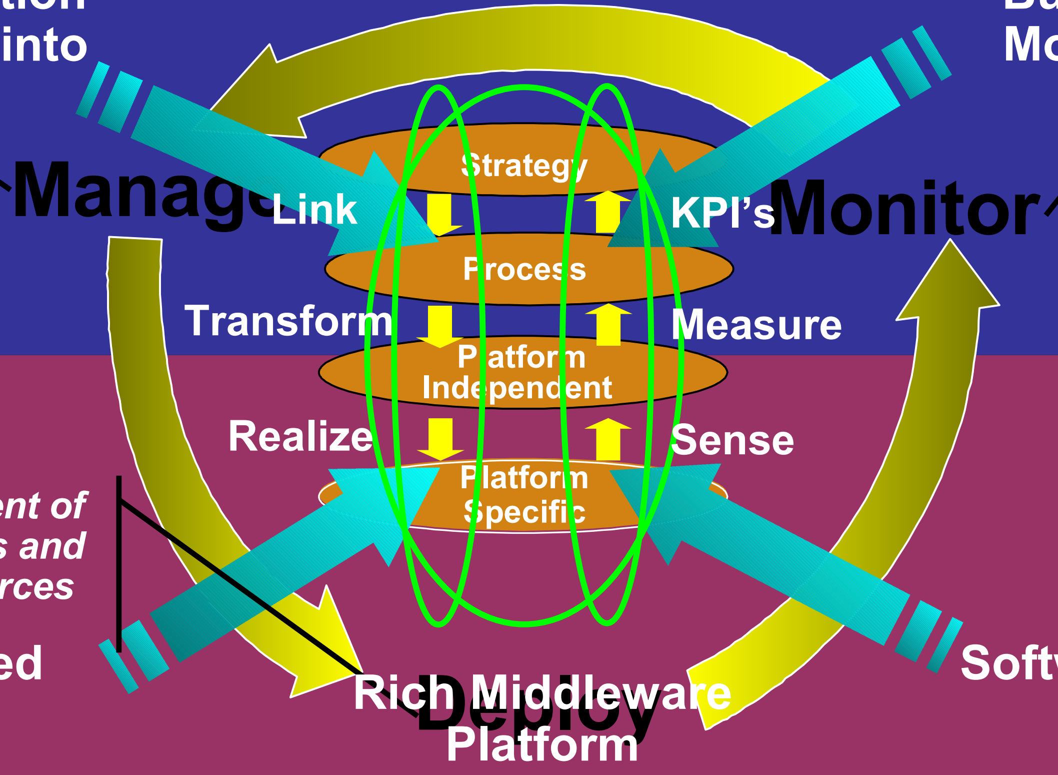
Business applications will be deployed, monitored and managed through the manipulation of multi-level models

Componentization of businesses into services

Flexibly transform business

Business Modeling

Real-time visib of business



Business Domain

IT Domain

Rapid deployment of enterprise apps and resources

Service Oriented Architecture

Software Modeling

Value: Accurately and reliably capture and translate business intent into IT solutions

A Simplified Example of Services

Business Process

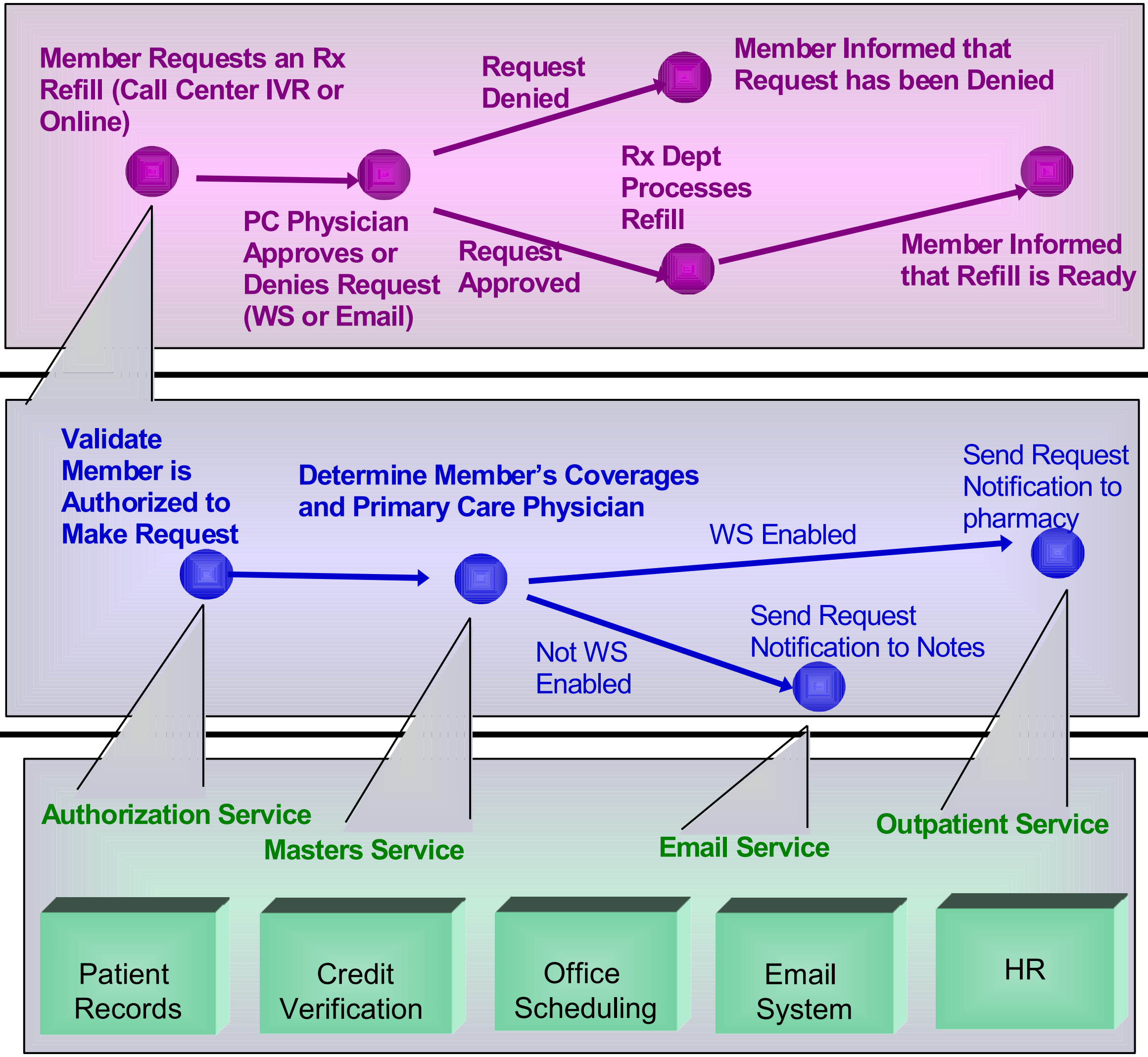
- long running
- one or more persons interacting
- multiple valid business process states
- alternative workflows for non-normal conditions

Services

- short term, non-interactive
- one change of business state
- consumes one or more enterprise service
- targeted level of service reuse
- loose coupling important
- may require compensating transactions

Components

- collaborations to implement a single Web Service
- collaborating apps encapsulated via Web Services
- Performance favored over loose coupling


Member Requests an Rx Refill (Call Center IVR or Online)
Request Denied
Member Informed that Request has been Denied
Rx Dept Processes Refill
PC Physician Approves or Denies Request (WS or Email)
Request Approved
Member Informed that Refill is Ready
Validate Member is Authorized to Make Request
Determine Member's Coverages and Primary Care Physician
Send Request Notification to pharmacy
WS Enabled
Send Request Notification to Notes
Not WS Enabled
Authorization Service
Masters Service
Email Service
Outpatient Service
Patient Records
Credit Verification
Office Scheduling
Email System
HR

A Simplified Example of Services

Business Process

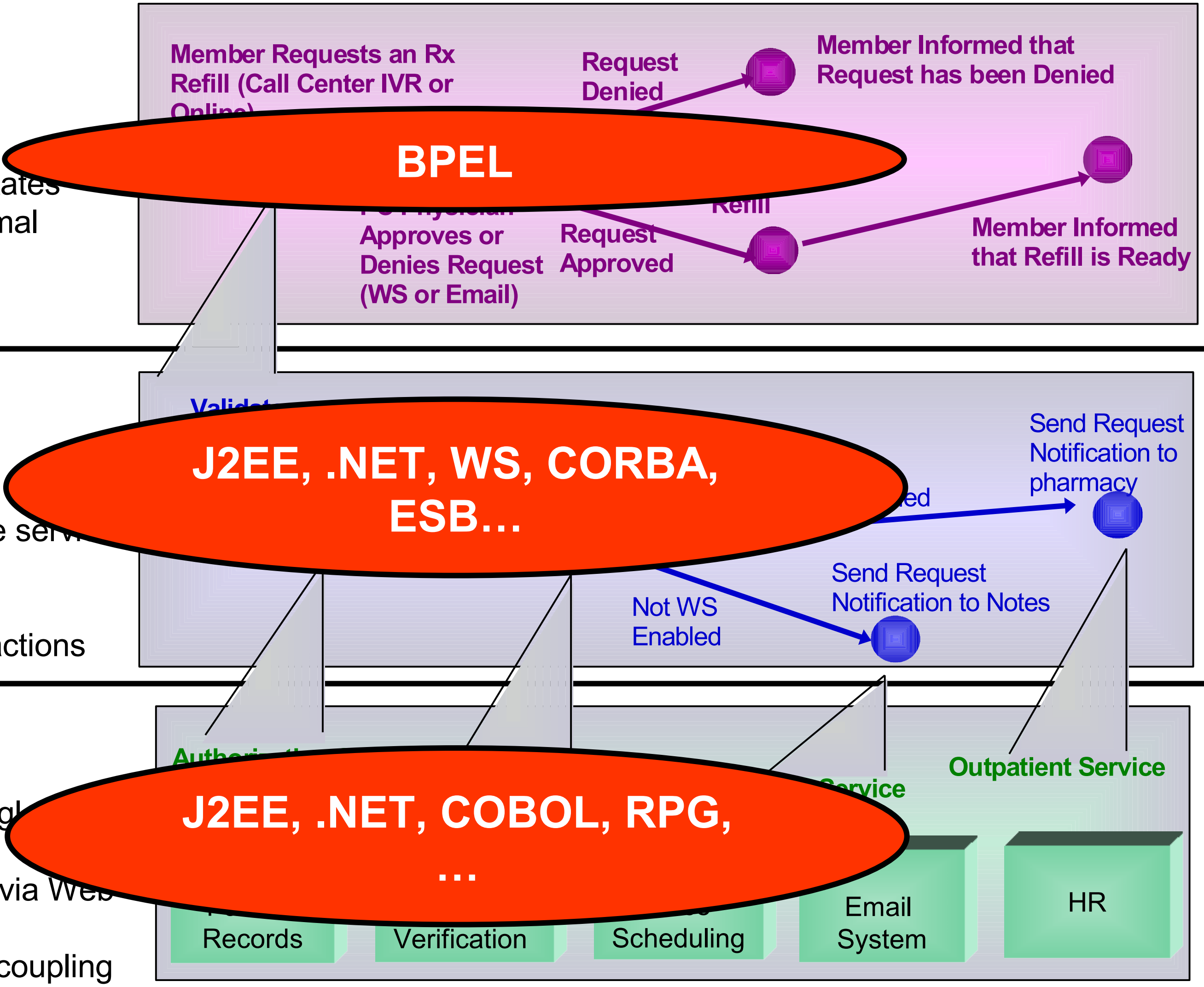
- long running
- one or more persons interacting
- multiple valid business process states
- alternative workflows for non-normal conditions

Services

- short term, non-interactive
- one change of business state
- consumes one or more enterprise serv
- targeted level of service reuse
- loose coupling important
- may require compensating transactions

Components

- collaborations to implement a singl Service
- collaborating apps encapsulated via Web Services
- Performance favored over loose coupling


Member Requests an Rx Refill (Call Center IVR or Online)
Request Denied
Member Informed that Request has been Denied
Refill
PCP Physician Approves or Denies Request (WS or Email)
Request Approved
Member Informed that Refill is Ready
BPEL
Validat
Send Request Notification to pharmacy
ed
Not WS Enabled
Send Request Notification to Notes
J2EE, .NET, WS, CORBA, ESB…
Authorization
Service
Outpatient Service
Records
Verification
Scheduling
Email System
HR
J2EE, .NET, COBOL, RPG, …

Key roles in service-oriented design and development of service-oriented applications

Business Analyst

- **Model the business**
  - Understand business requirements
  - Analyze and develop process models
  - Identify optimum process models to drive services design

Software Architect

- **Design the services architecture**
  - Model and refine the services architecture
  - Identify new services needed and existing assets to re-use
  - Generate services specifications

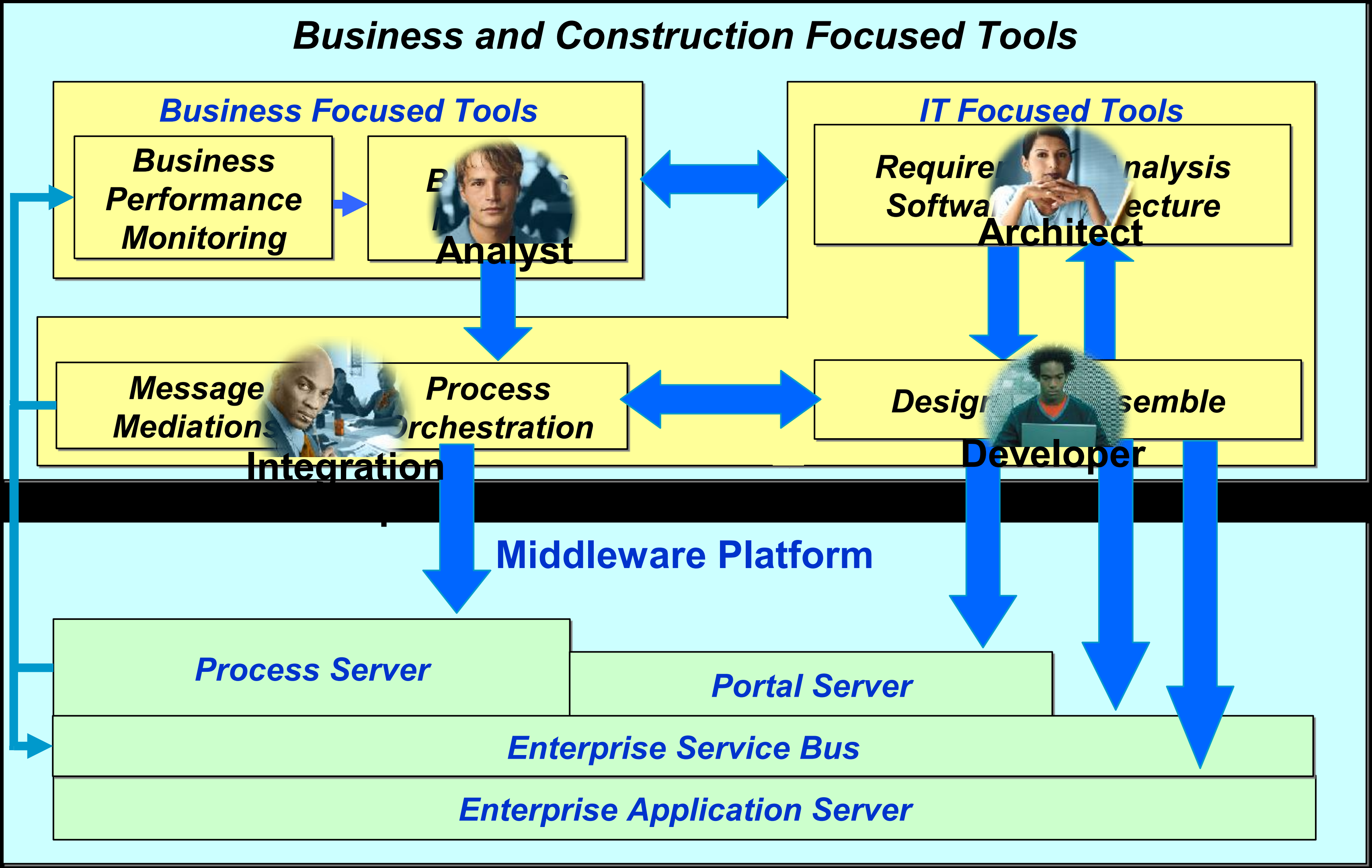
Developer

- **Construct the services**
  - Implement new services & repurpose existing assets as services
  - Create UI for access via Web or Portal
  - Validate and test services

Integration Specialist

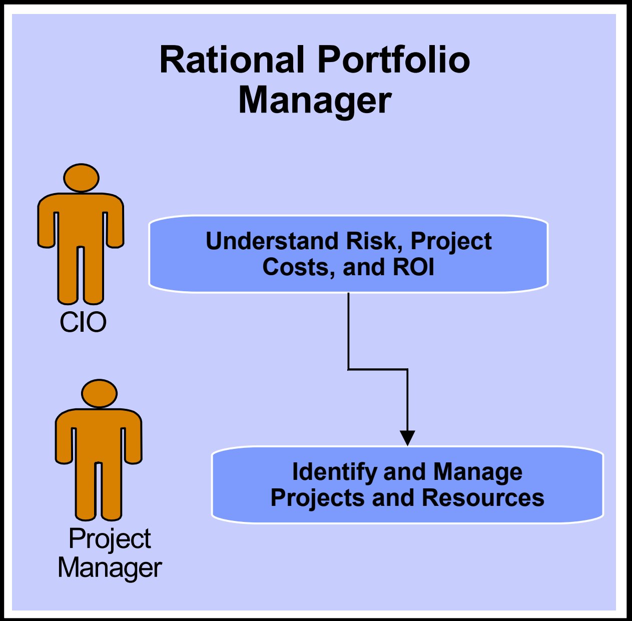
- **Assemble and deploy composite application**
  - View the process model
  - Choreograph the services
  - Assemble and deploy

Business Driven Development Scenario



Manage Projects and Portfolios

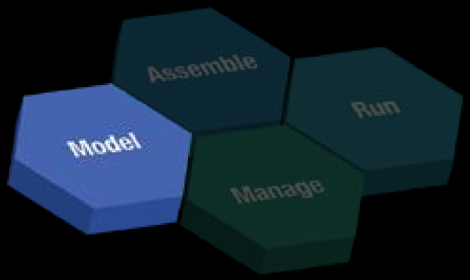
- Prioritize proposed, existing and under-construction services based on business priority, risk and return
- Track service level financials
- Provide deep insight into SOA development
- Manage SOA project-team dependencies
- Forecast demand for service creation and updates
- Understand the cost of SOA creation, operations and maintenance



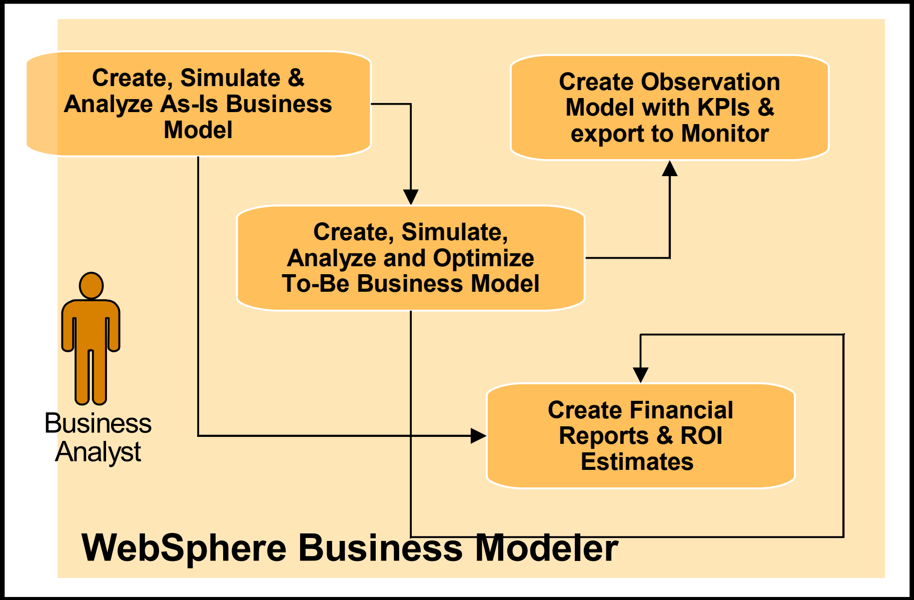
Rational Portfolio Manager

CIO
Understand Risk, Project Costs, and ROI
Project Manager
Identify and Manage Projects and Resources

Model the Business



- Discover and design key business processes
- Determine and allocate required resources
- Model the business organization & roles organizational units can play
- Determination of any other process/tasks (services)
  that must be provided by others


Create, Simulate & Analyze As-Is Business Model

Create Observation Model with KPIs & export to Monitor

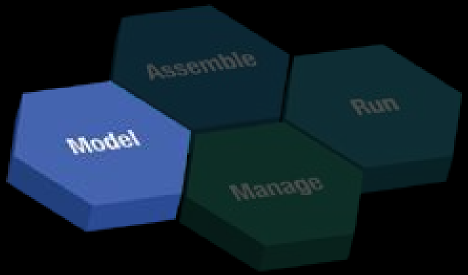
Create, Simulate, Analyze and Optimize To-Be Business Model

Create Financial Reports & ROI Estimates

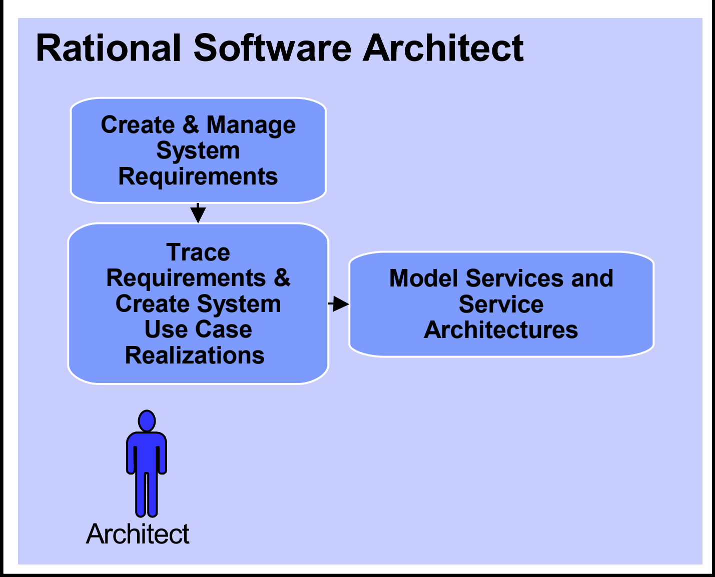
Business Analyst

WebSphere Business Modeler

Architect and Design the Services



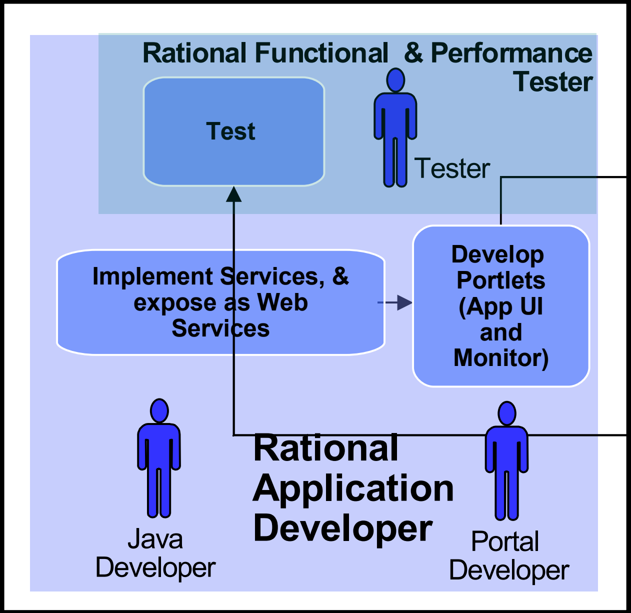
- Trace enterprise requirements to business processes and service implementations
- Define detailed system requirements and service implementations
- Architect and design the service implementations


Rational Software Architect

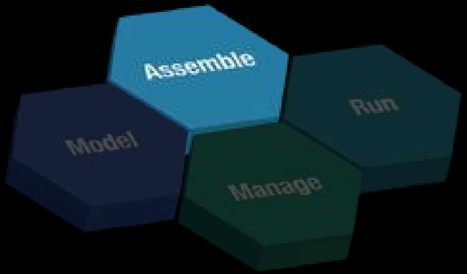
Create & Manage System Requirements
Trace Requirements & Create System Use Case Realizations
Model Services and Service Architectures
Architect

Construct and Test the Services

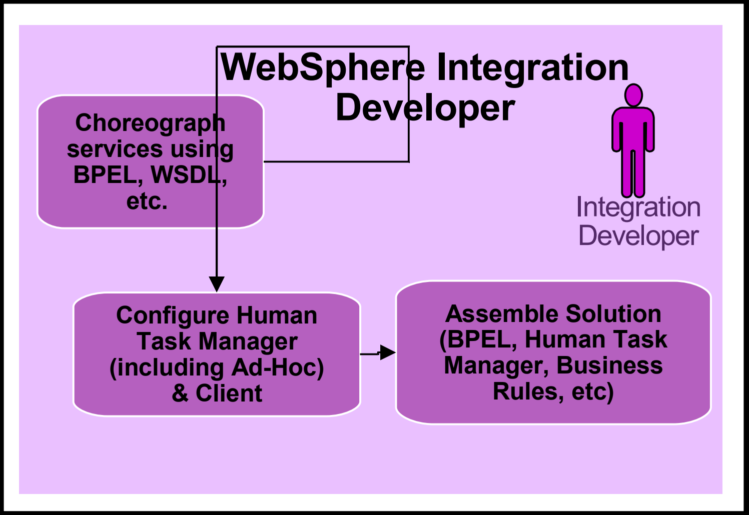
- **Build new services from scratch or enable existing applications for WS-I compliance**
- **Discover and consume existing services**
- **Test functionality**
- **Test performance**


Rational Functional & Performance Tester
Test
Tester
Implement Services, & expose as Web Services
Develop Portlets (App UI and Monitor)
Rational Application Developer
Java Developer
Portal Developer

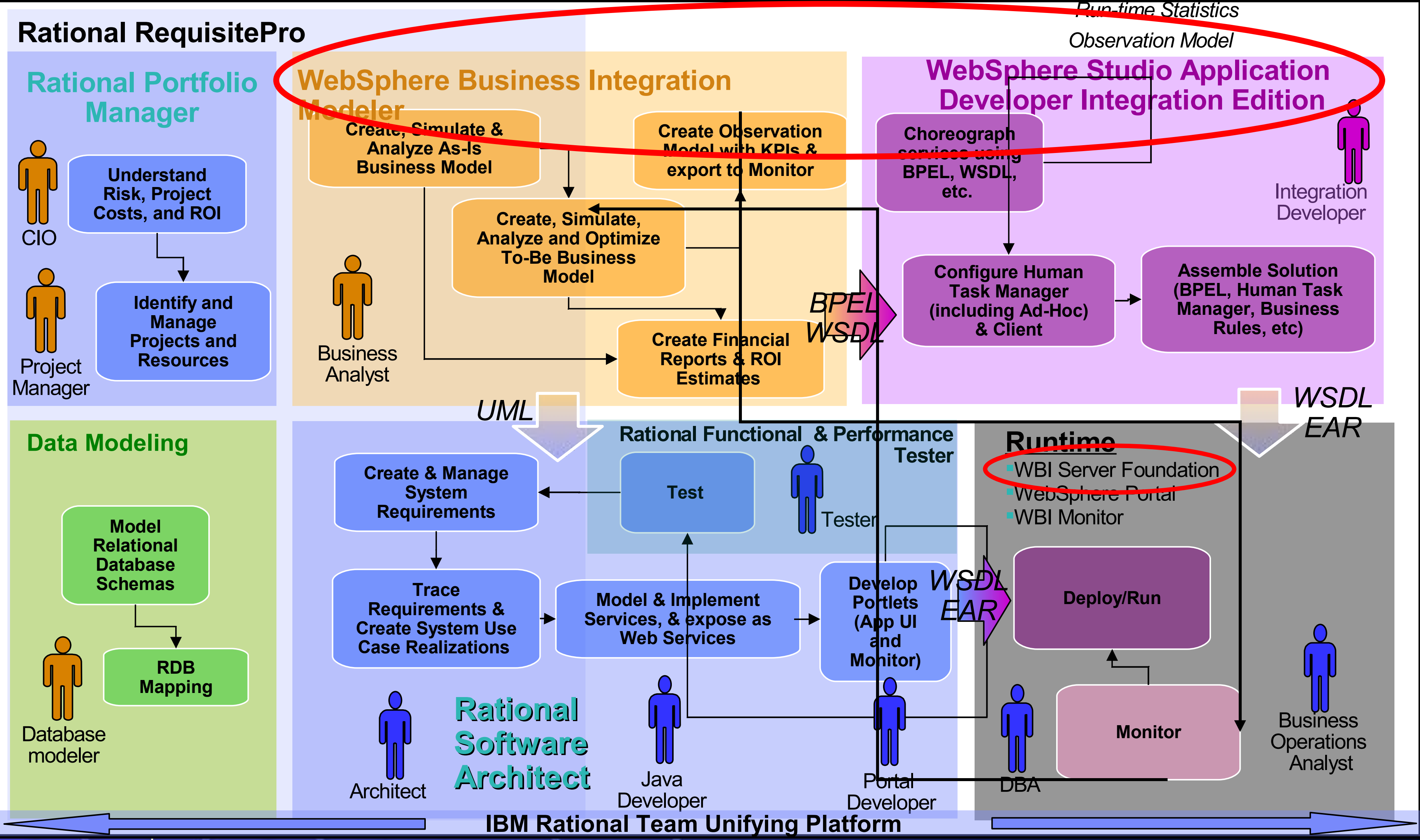
Assemble composite application



- **Implement business processes designed by Business Analysts**
  - Plug in Services
  - Plug in Human Activities (Staff)
- **Test composite application**


WebSphere Integration Developer
Choreograph services using BPEL, WSDL, etc.
Integration Developer
Configure Human Task Manager (including Ad-Hoc) & Client
Assemble Solution (BPEL, Human Task Manager, Business Rules, etc)

Business–Driven Development: The Big Picture



Run-time Statistics

Observation Model

Rational RequisitePro

Rational Portfolio Manager

CIO

Understand Risk, Project Costs, and ROI

Identify and Manage Projects and Resources

Project Manager

WebSphere Business Integration Modeler

Create, Simulate & Analyze As-Is Business Model

Create Observation Model with KPIs & export to Monitor

Create, Simulate, Analyze and Optimize To-Be Business Model

Create Financial Reports & ROI Estimates

Business Analyst

BPEL WSDL

WebSphere Studio Application Developer Integration Edition

Choreograph services using BPEL, WSDL, etc.

Integration Developer

Configure Human Task Manager (including Ad-Hoc) & Client

Assemble Solution (BPEL, Human Task Manager, Business Rules, etc)

WSDL EAR

UML

Data Modeling

Model Relational Database Schemas

RDB Mapping

Database modeler

Rational Functional & Performance Tester

Create & Manage System Requirements

Test

Tester

Trace Requirements & Create System Use Case Realizations

Model & Implement Services, & expose as Web Services

Develop Portlets (App UI and Monitor)

WSDL EAR

Architect

Rational Software Architect

Java Developer

Portal Developer

Runtime

- WBI Server Foundation
- WebSphere Portal
- WBI Monitor

Deploy/Run

Monitor

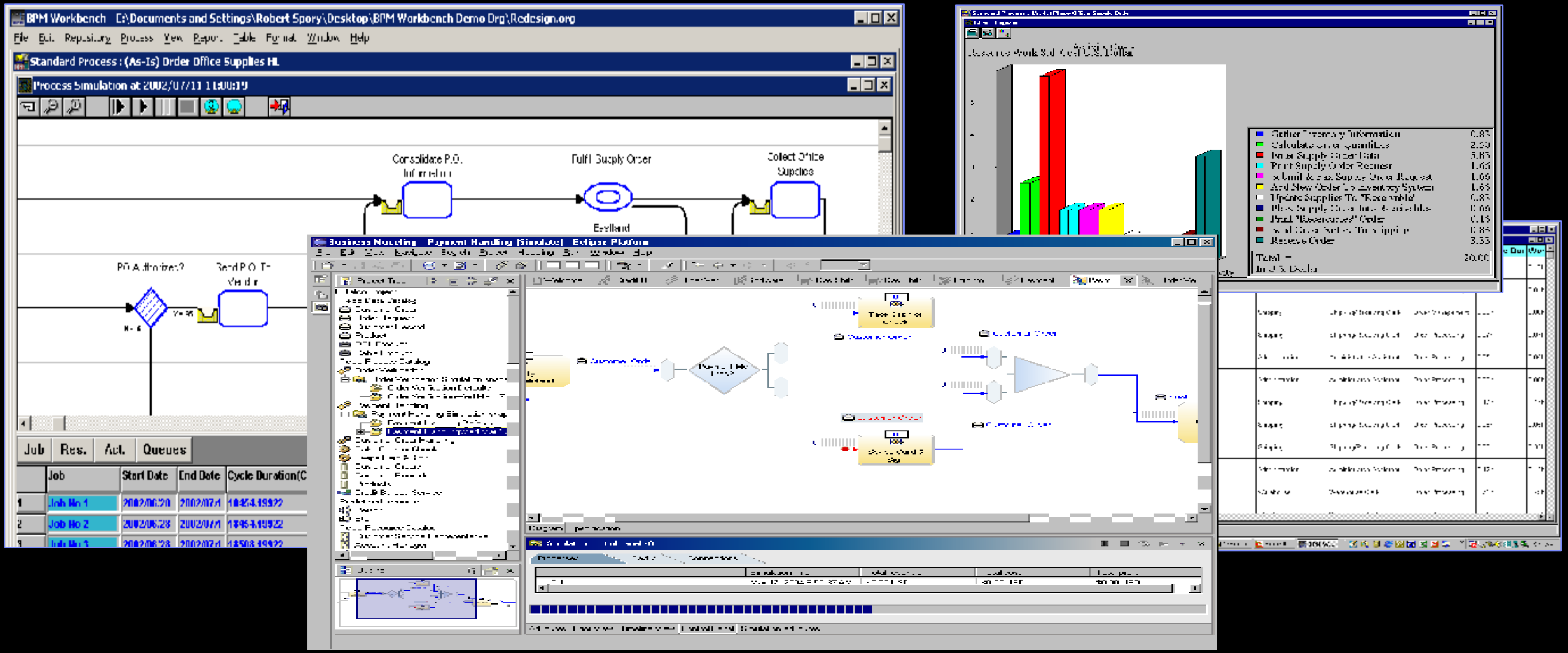
DBA

Business Operations Analyst

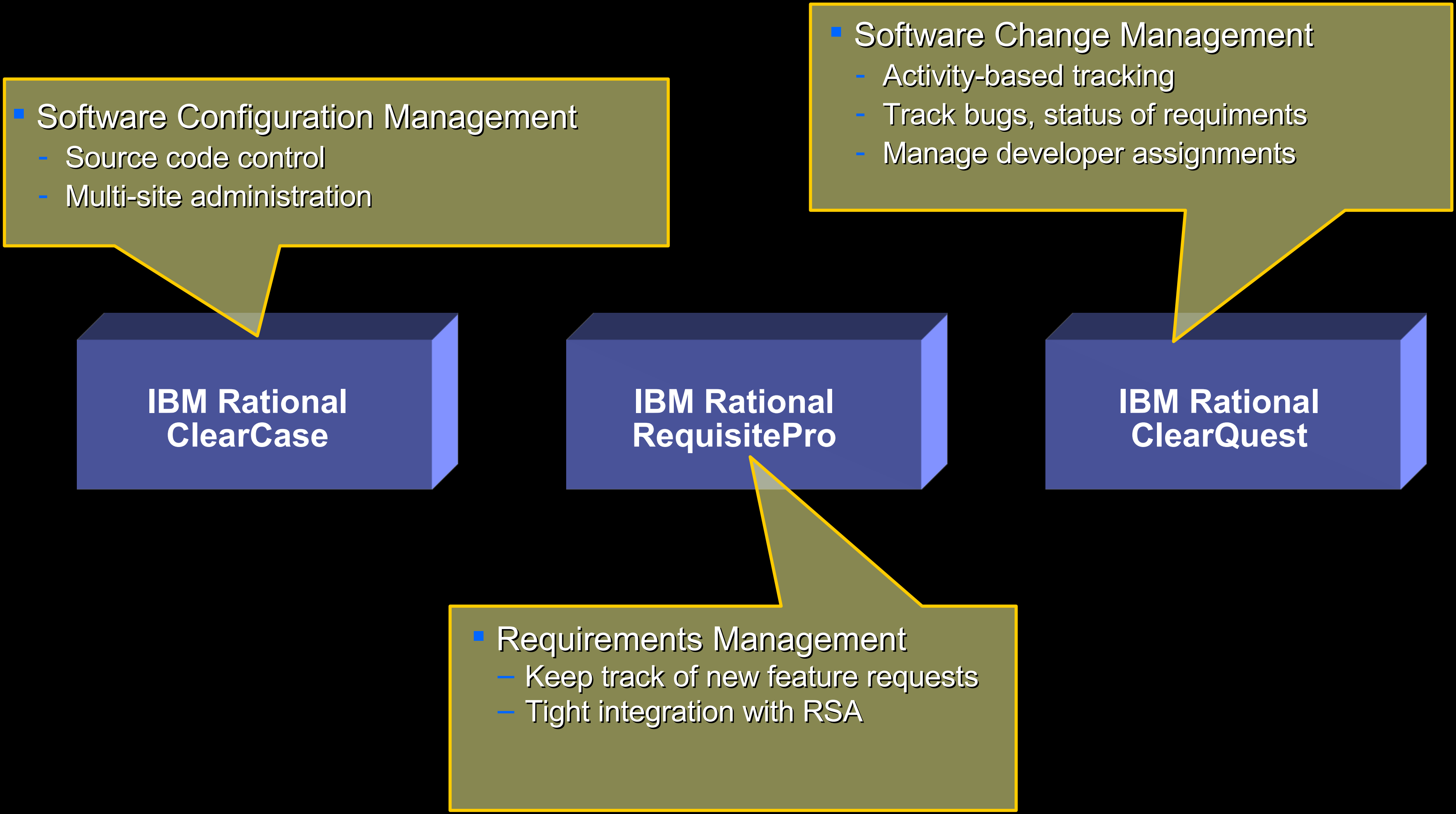
IBM Rational Team Unifying Platform

WebSphere Business Modeler

- Graphically design processes and quickly redesign across people, partners and applications
- “What-if” simulation of operations to optimize and project business benefits
- Fast start to deployment—generates code from model


BPM Workbench C:\Documents and Settings\Robert Spory\Desktop\BPM Workbench Demo Org\Redesign.org
File Edit Repository Process View Report Table Format Window Help
Standard Process : (As-Is) Order Office Supplies HL
Process Simulation at 2002/07/11 11:00:19
Consolidate P.O. Information
Fulfil Supply Order
Collect Office Supplies
Bestland
PO Authorized?
Send P.O. To Vendor
Yes
No
Job
Res.
Act.
Queues
Job
Start Date
End Date
Cycle Duration(C
Job No 1
Job No 2
Job No 3
Business Modeling - Payment Handling [Simulate] - Eclipse Platform
Gather Inventory Information
Calculate Order Quantities
Print Supply Order Request
Add New Order To Inventory System
Update Supplies To "Receivable"
Print "Received" Order
Receive Order
Total
0.83
2.50
5.83
1.66
1.66
1.65
0.83
0.66
0.16
0.84
3.33
20.00

Change, requirements and asset management



- Software Configuration Management
  - Source code control
  - Multi-site administration

- Software Change Management
  - Activity-based tracking
  - Track bugs, status of requiments
  - Manage developer assignments

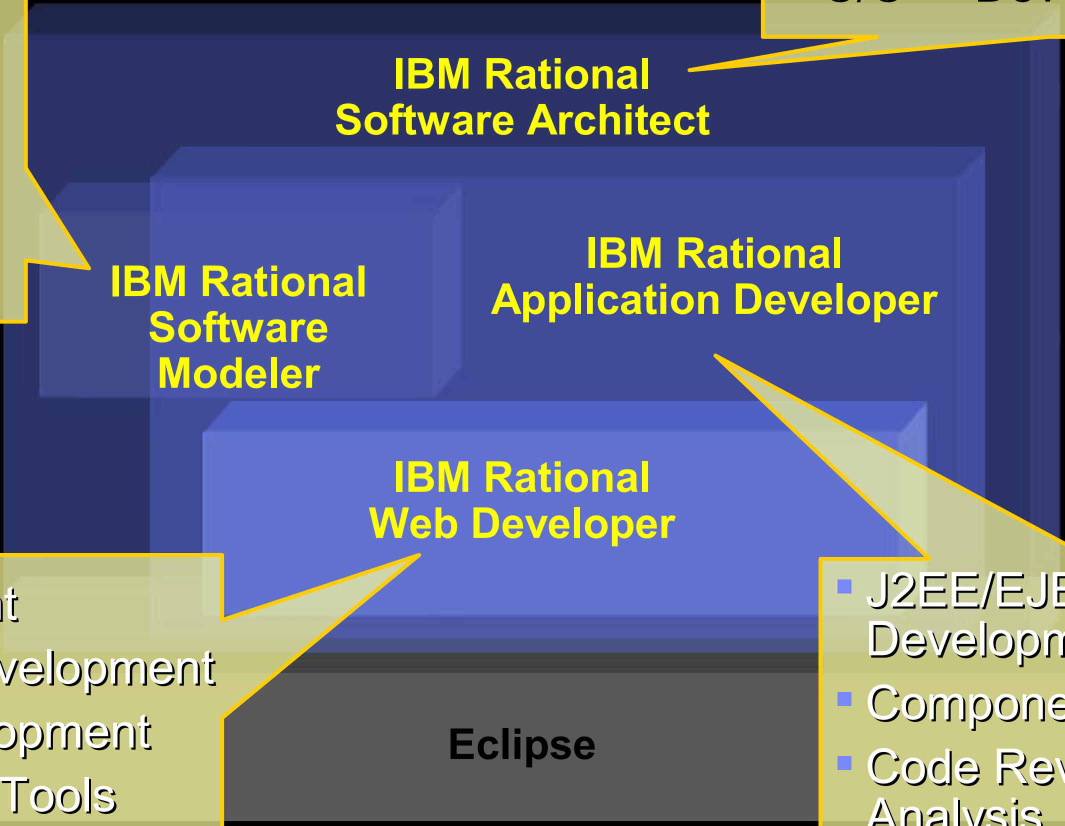
IBM Rational ClearCase

IBM Rational RequisitePro

IBM Rational ClearQuest

- Requirements Management
  - Keep track of new feature requests
  - Tight integration with RSA

Design and construction tools


IBM Rational Software Architect
IBM Rational Software Modeler
IBM Rational Application Developer
IBM Rational Web Developer
Eclipse

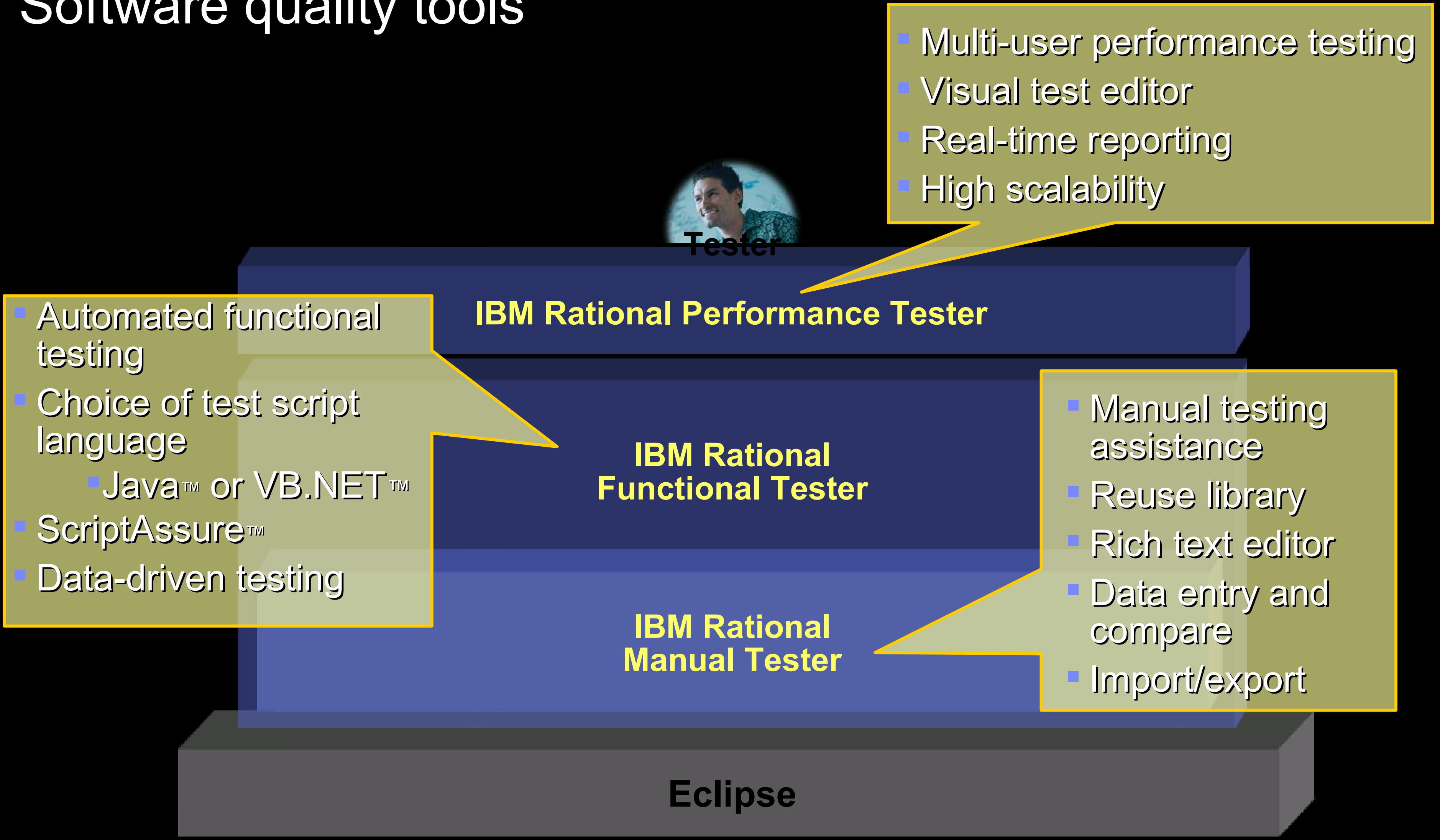
- UML Language Transforms
- Structural Review & Control
- C/C++ Development Tools

- UML 2.0
- Pattern/ Transform Authoring
- Reusable Asset Browser

- Web Development
- Web Services Development
- Rich Client Development
- XML & Database Tools
- Java Generation Tools
- Unit Test

- J2EE/EJB & Portal Development
- Component Testing
- Code Review & Runtime Analysis
- UML Visual Editors
- Configuration Management

Software quality tools



- Multi-user performance testing
- Visual test editor
- Real-time reporting
- High scalability

Tester

IBM Rational Performance Tester

- Automated functional testing
- Choice of test script language
  - Java™ or VB.NET™
- ScriptAssure™
- Data-driven testing

IBM Rational Functional Tester

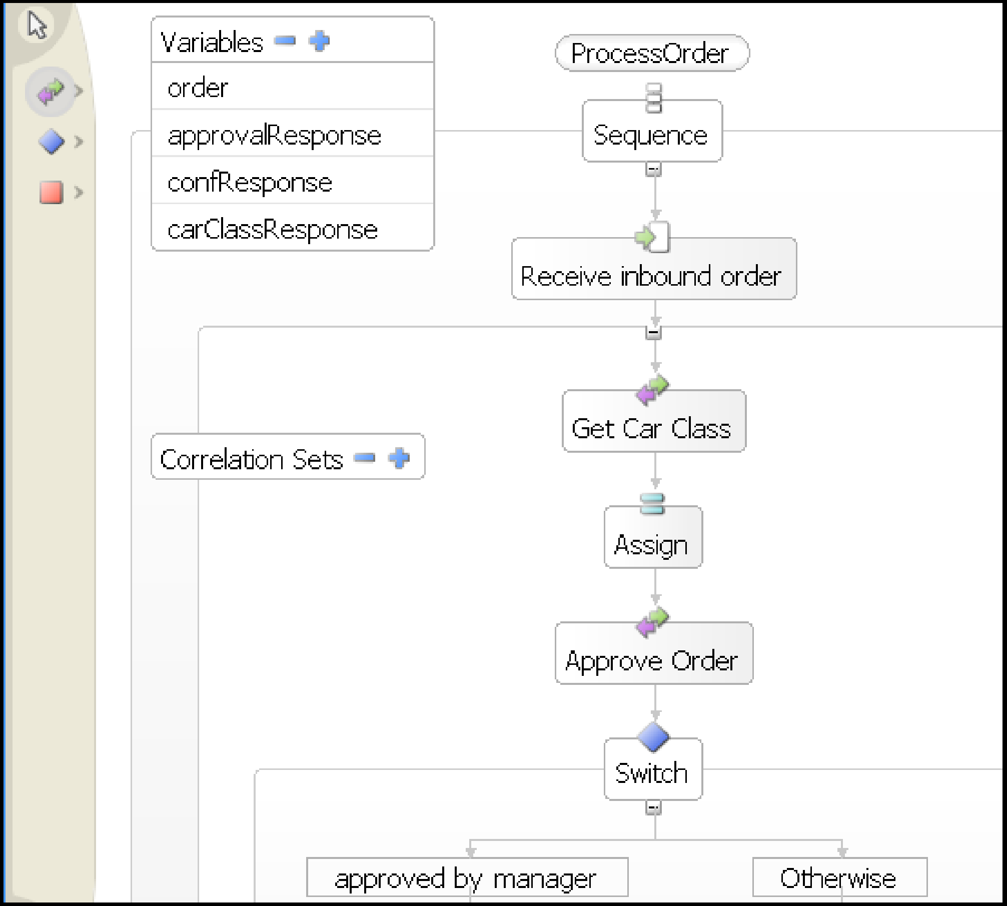
- Manual testing assistance
- Reuse library
- Rich text editor
- Data entry and compare
- Import/export

IBM Rational Manual Tester

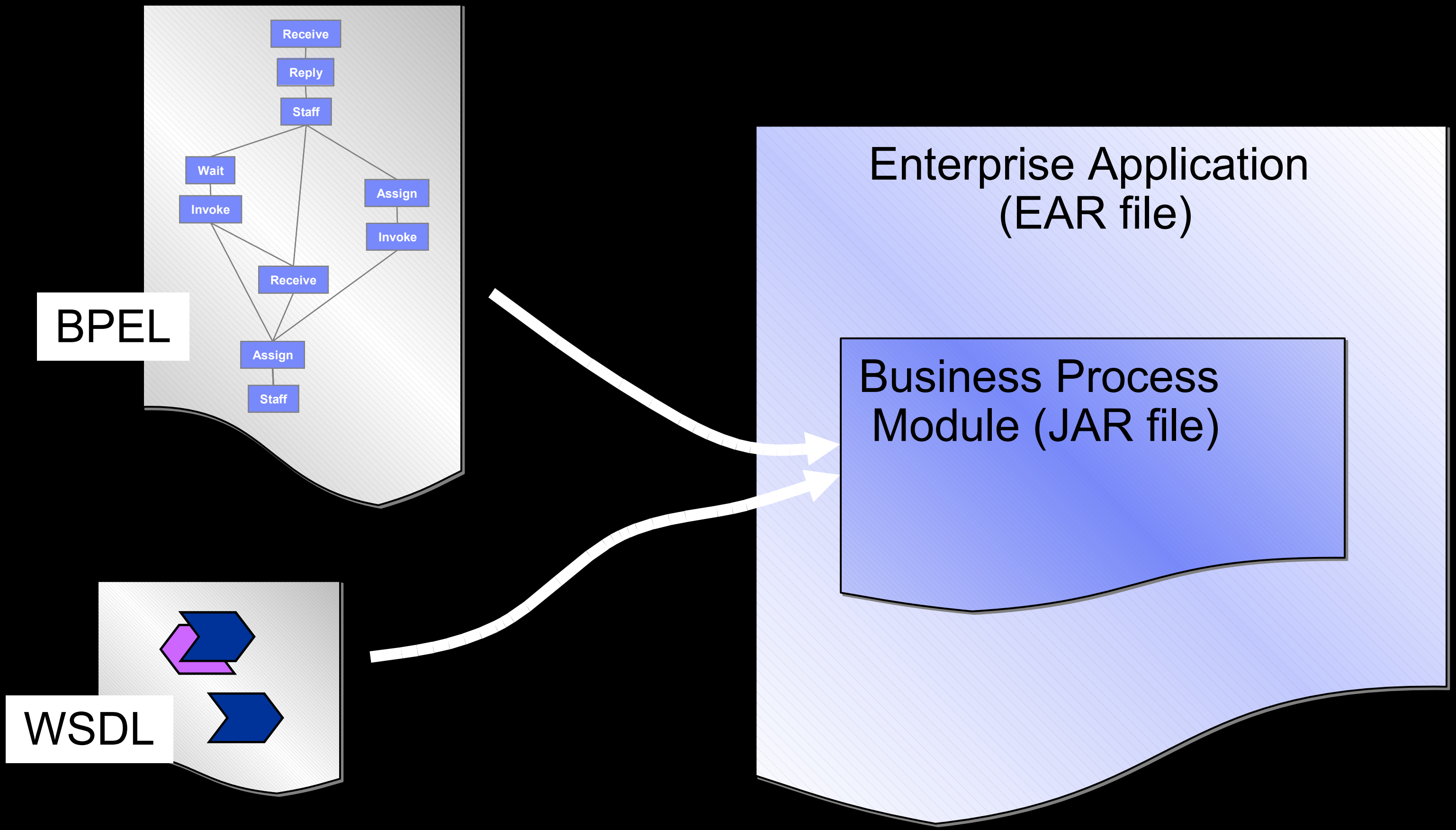
Eclipse

Business Process Choreography

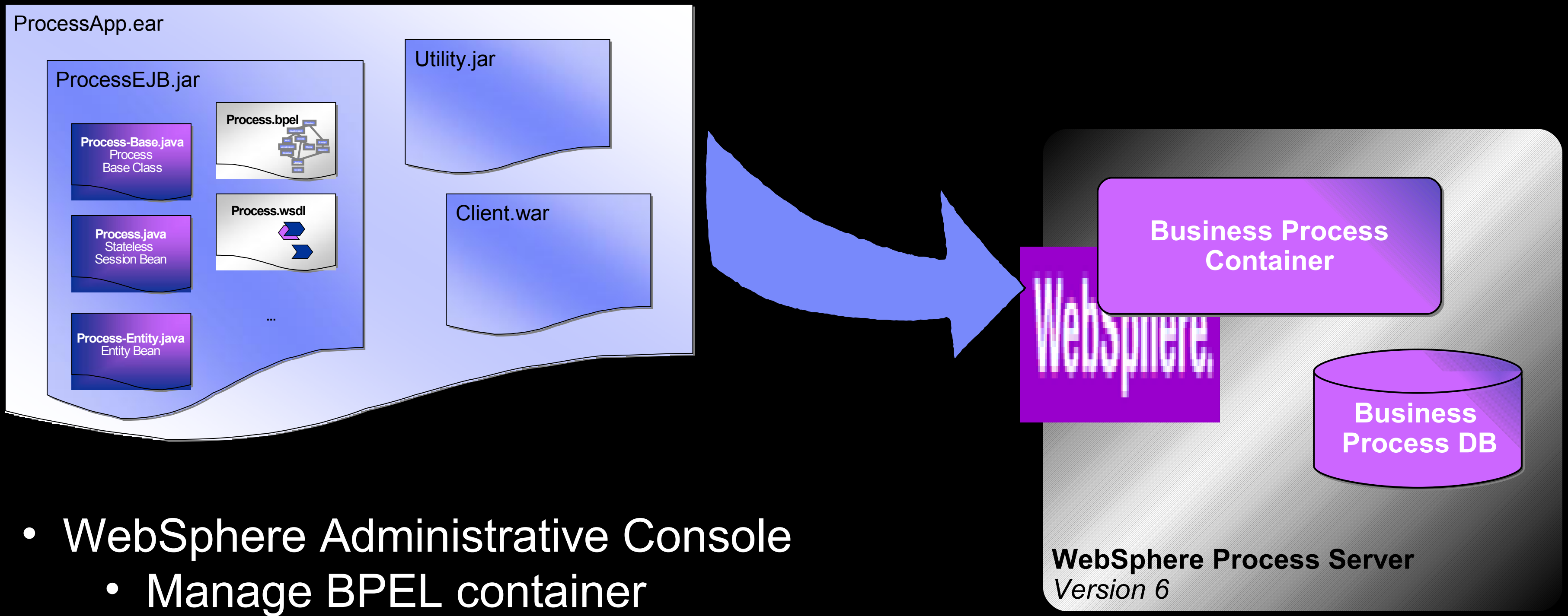
- Business Process Choreography is linking services together to form a deployable business processes:
  - Deployable process model (WS-BPEL) derived from business process model designed by a Business Analyst
  - Both Flow and Event based Business Process can be modeled
  - Choreography includes automated and human based services
  - Specify IT KPI’s



Composing a Business Process Application

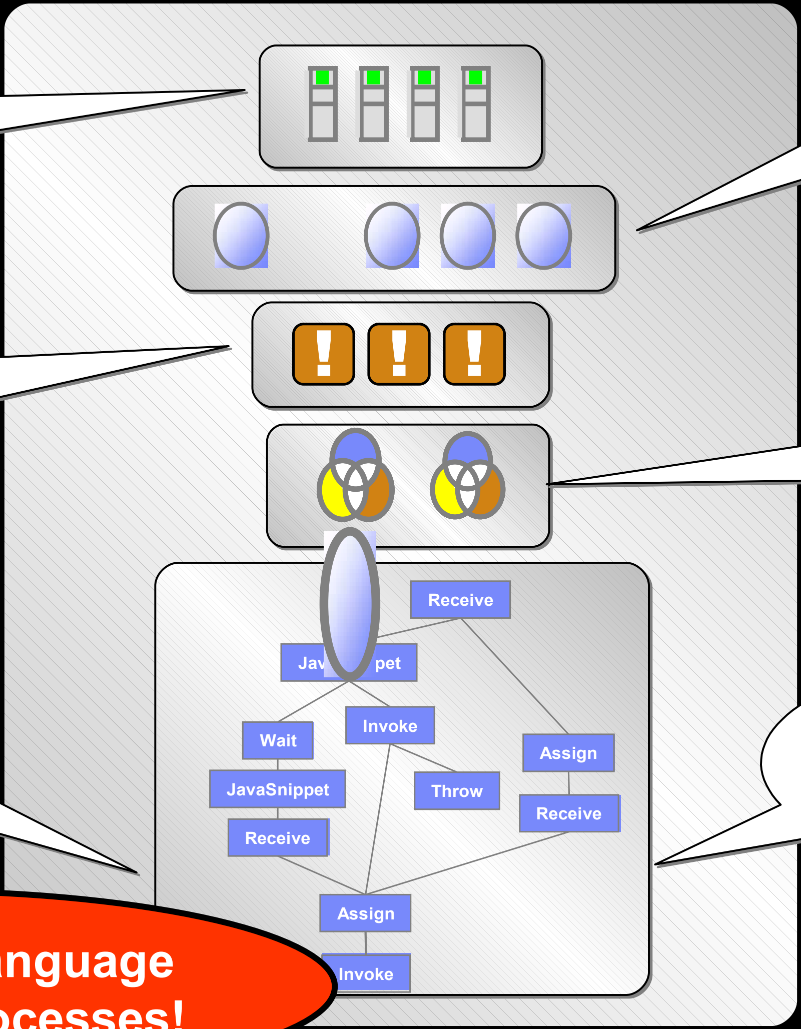


Deployment of the Business Process Application



- WebSphere Administrative Console
  - Manage BPEL container
  - Install/uninstall BPEL applications
  - Start/stop BPEL applications

Elements of a BPEL Process



PartnerLinks placeholders for process callers and service providers

Variables hold data used in the business process

FaultHandlers enclose activities that are performed in cases of error

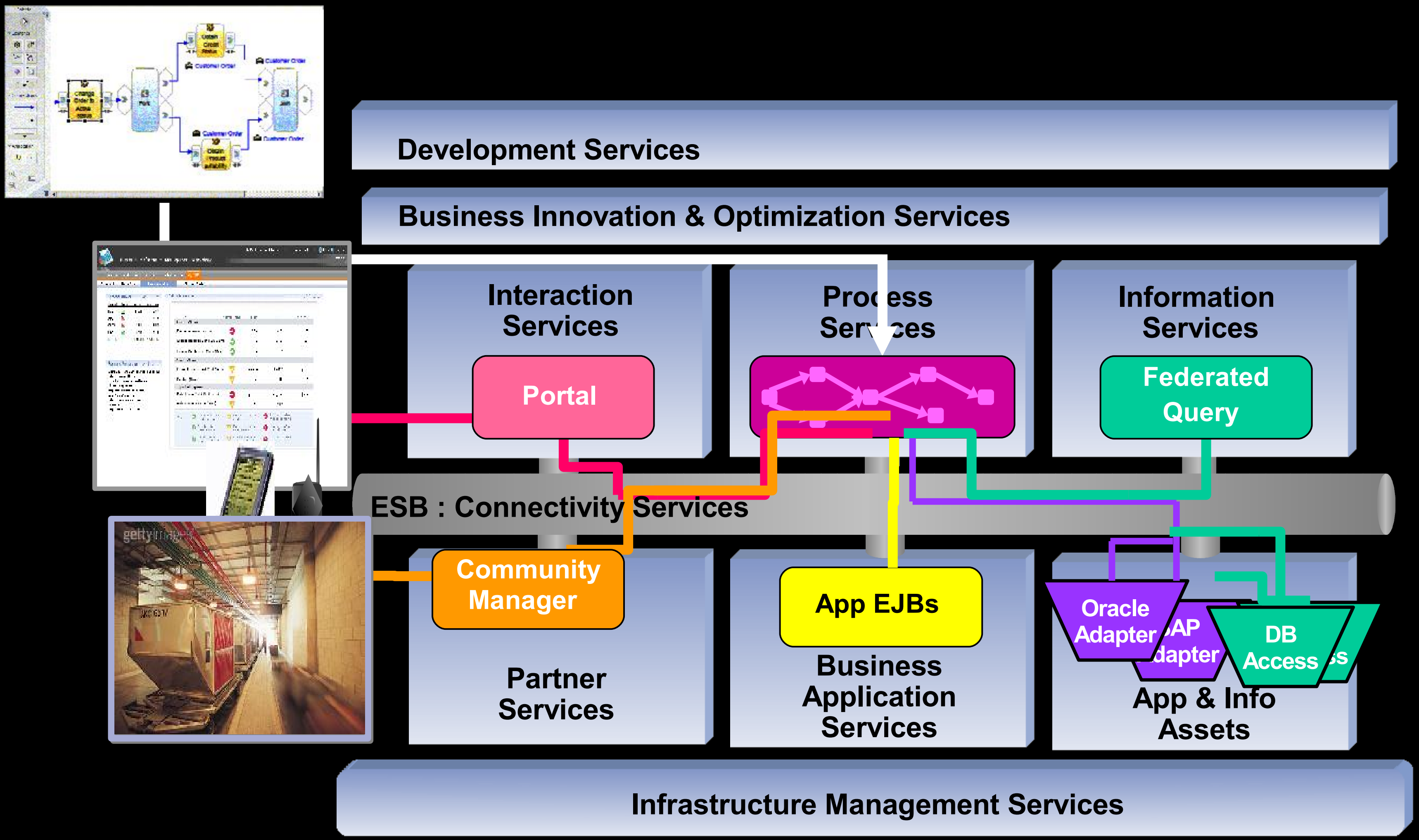
CorrelationSets support process instance identification

Activities subtasks of the process

Control Links define the process' control flow

Programming language for Business Processes!

Composite Application Development through SOA

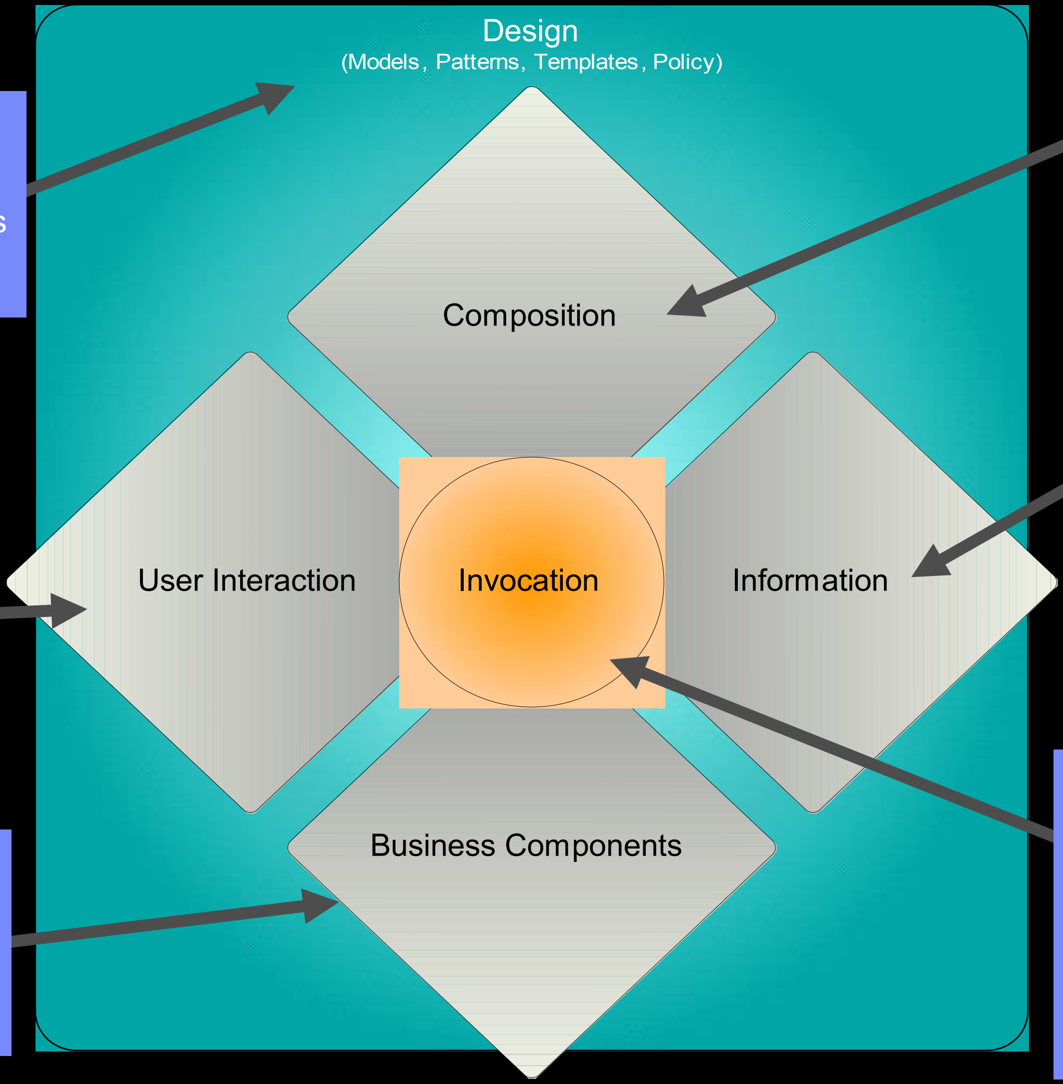


Assembling with the SOA Programming Model

A Complex Programming Model Today

- Visio Diagrams
- Books
- Code Samples
- J2EE Deployment Descriptors
- Java Property Files
- XML

- FDML Workflow
- EJB SessionBeans



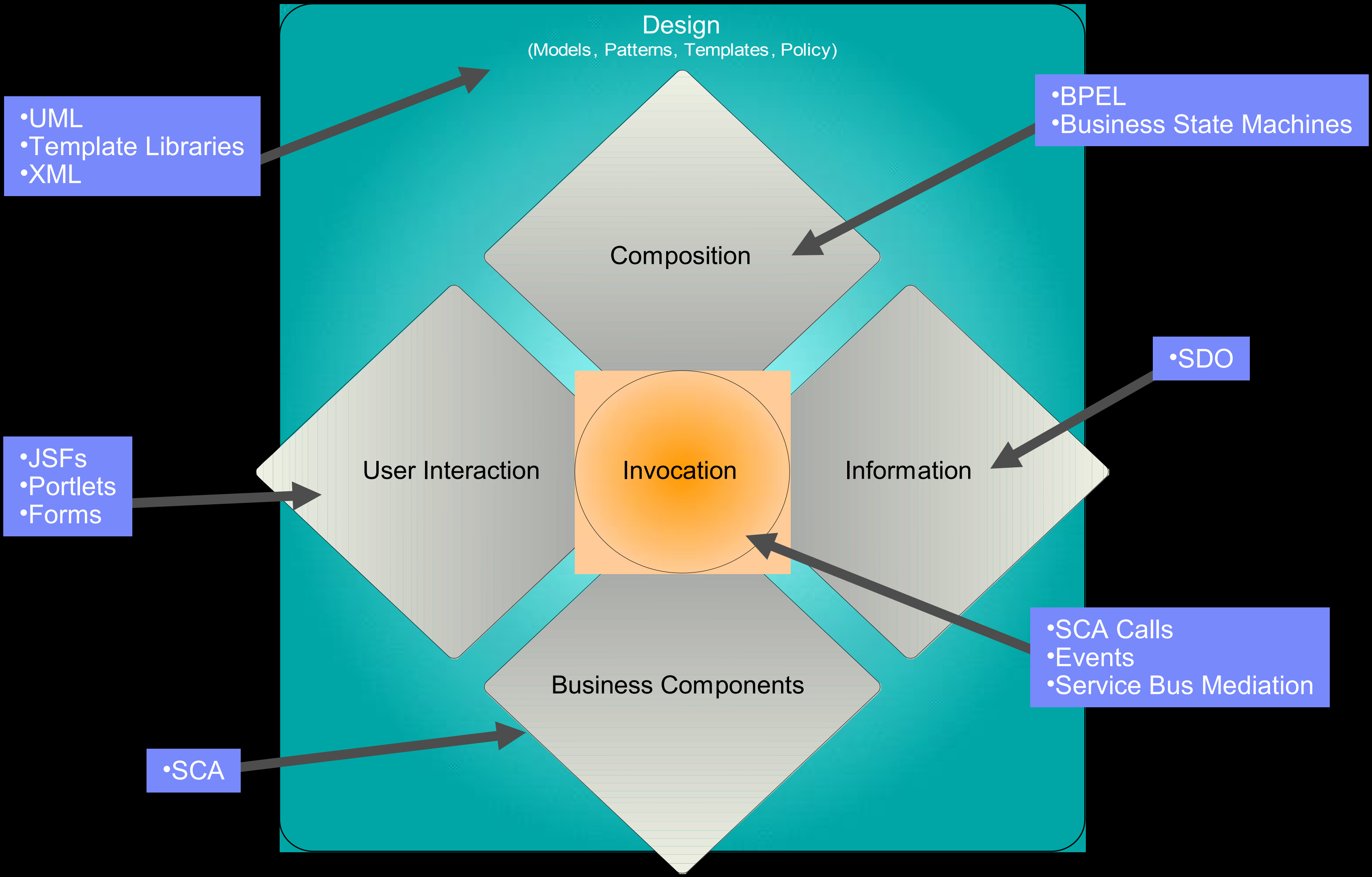
- JSPs / Servlets
- Struts / Tiles
- Portlets
- Click-to-Action
- XSLT
- Swing / AWT / SWT
- Native thick client
- 3270 / 5250

- JDBC Row Set
- Java Bean
- JAXB Object
- JMS/MQ Message
- JCA Data
- Data Access Beans
- SAAJ
- EMF Data
- XML DOM
- WICS BOs/GBOs

- EJB SessionBeans
- POJOs
- CICS/IMS Transactions
- MQ Applications
- Stored Procedures
- WICS Collaborations

- MQ API
- JMS
- J2CA Adapters
- WBI Adapters
- WICS Adapters
- RMI/IIOP
- JAX-RPC
- WS-GW Handlers
- WSRP

We Need A Simplified SOA Programming Model

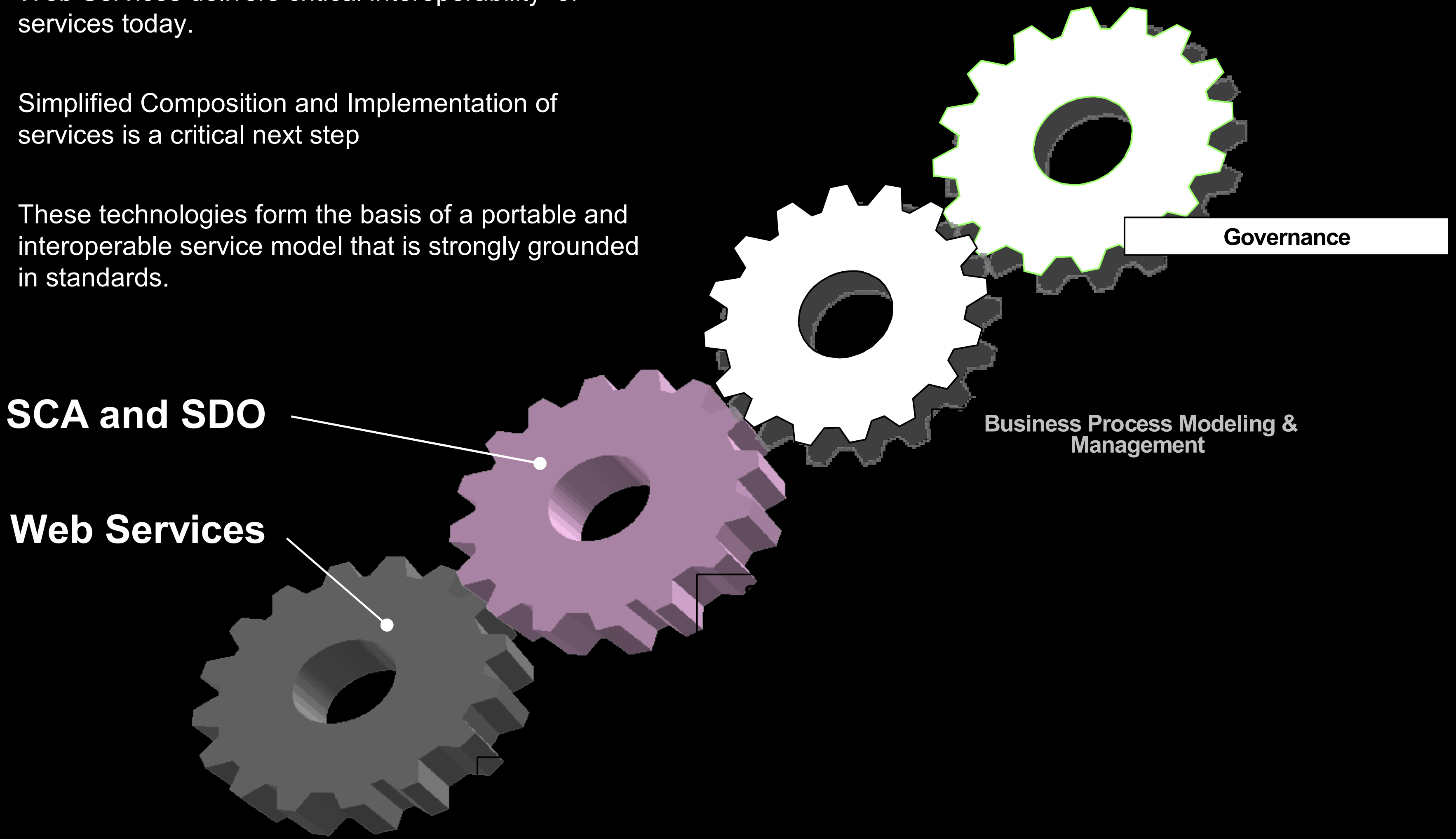


SCA and SDO

- A Language Neutral Assembly Model specification to simplify the composition and development of Business Services called: "Service Component Architecture"
- A Java Language specification for implementing SCA service components
- A C++ Language specification for implementing SCA service components
- A Java Language Service Data Objects specification describing a common rendering methodology for data exchange between clients and services
- A C++ Language Service Data Objects specification describing a common rendering methodology for data exchange between clients and services

SOA Infrastructure Standards Roadmap

- Web Services delivers critical interoperability of services today.
- Simplified Composition and Implementation of services is a critical next step
- These technologies form the basis of a portable and interoperable service model that is strongly grounded in standards.



Simplified Programming Model In a Nutshell

What are SCA and SDO?

- SCA provides a model for constructing and assembling networks of services
- SDO provides common access to data
- Simplifies the development and usage of services
- Provides a simplified programming experience for services and data

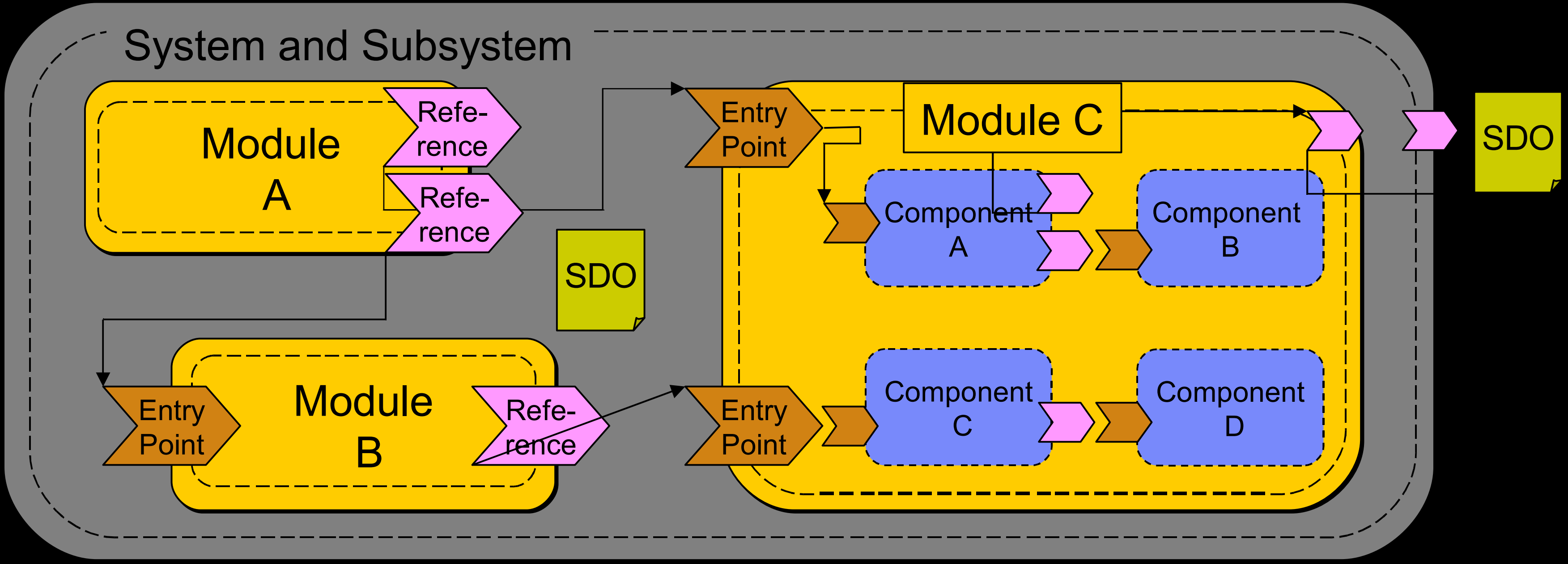
SCA Features

- Multi language support for Assembling Services, e.g. Java, WSDL, C++, PHP, …
- Runtime access to a diverse set of services, e.g. Web Service, JCA, JMS, Data, etc.
- Extensible, new interface, implementation, and bindings types
- First class support for secure, reliable, and transacted Web Services

SDO Features

- Simplifies access to data (query, process/update, and persist data)
- Supports Multiple Languages

Service Component Architecture

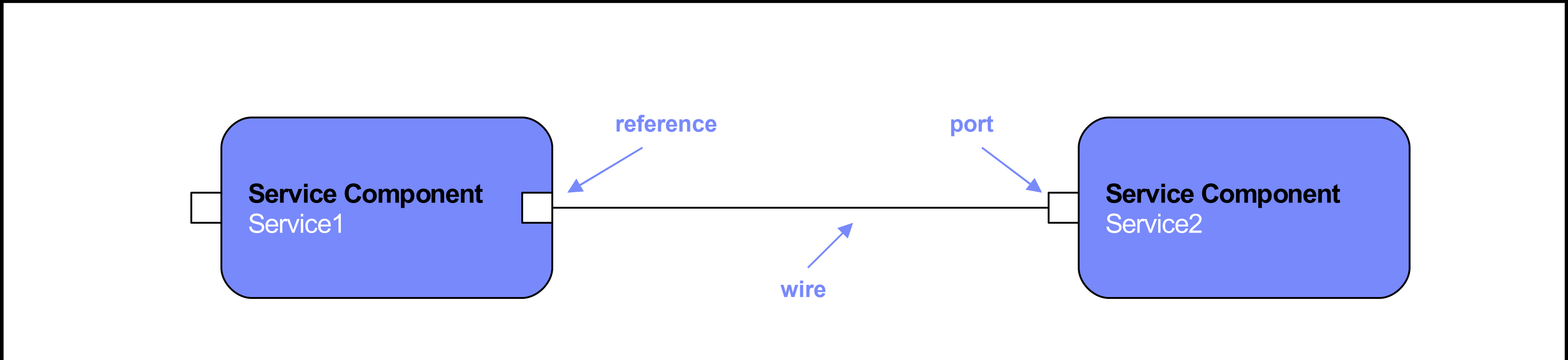


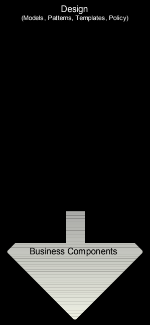
Design
(Models, Patterns, Templates, Policy)

Service Component Architecture

Business Components

- A service component defines a service
- A service supports zero or more ports
- Each port has a defined set of operations and messages that it supports
- A service uses zero or more services (or publishes to zero or more topics)
  - Each dependent service is represented in the service as a reference and qualified by an interface or message definition
- Service can be composed by wiring a service reference to a service port

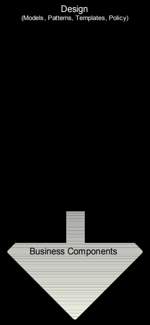




Service Ports and References

- Ports and References are typed
  - Can be typed in any number of different type-encoding languages
    - Java Interfaces
    - WSDL Port Types
    - UML
    - COBOL Copy Books
    - C/C++, PL/I, RPG procedure definitions
    - …. it’s extensible

- The type-encoding language does not (necessarily) dictate/indicate the implementation language of the component

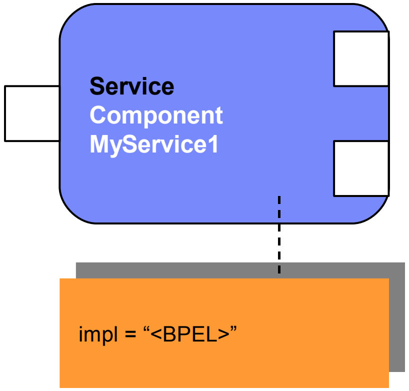


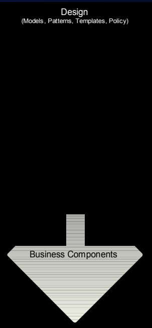
Component Definition vs. Component Implementation

- Services can be composed (wired) without regard for the service implementation
  - "A service is merely an abstraction that encapsulates a software function. The value of any abstraction is in reducing conceptual burden. A service oriented abstraction models only the details that are necessary and relevant in order to use the service. Detail that is not relevant – such as the service's specific technology underpinnings or its internal implementation—can be omitted from the model without impairing the programmer's ability to use the service effectively."

SWG SOA: Programming Model and Architectural Overview

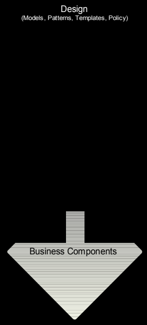
- Nonetheless, a component does have an implementation





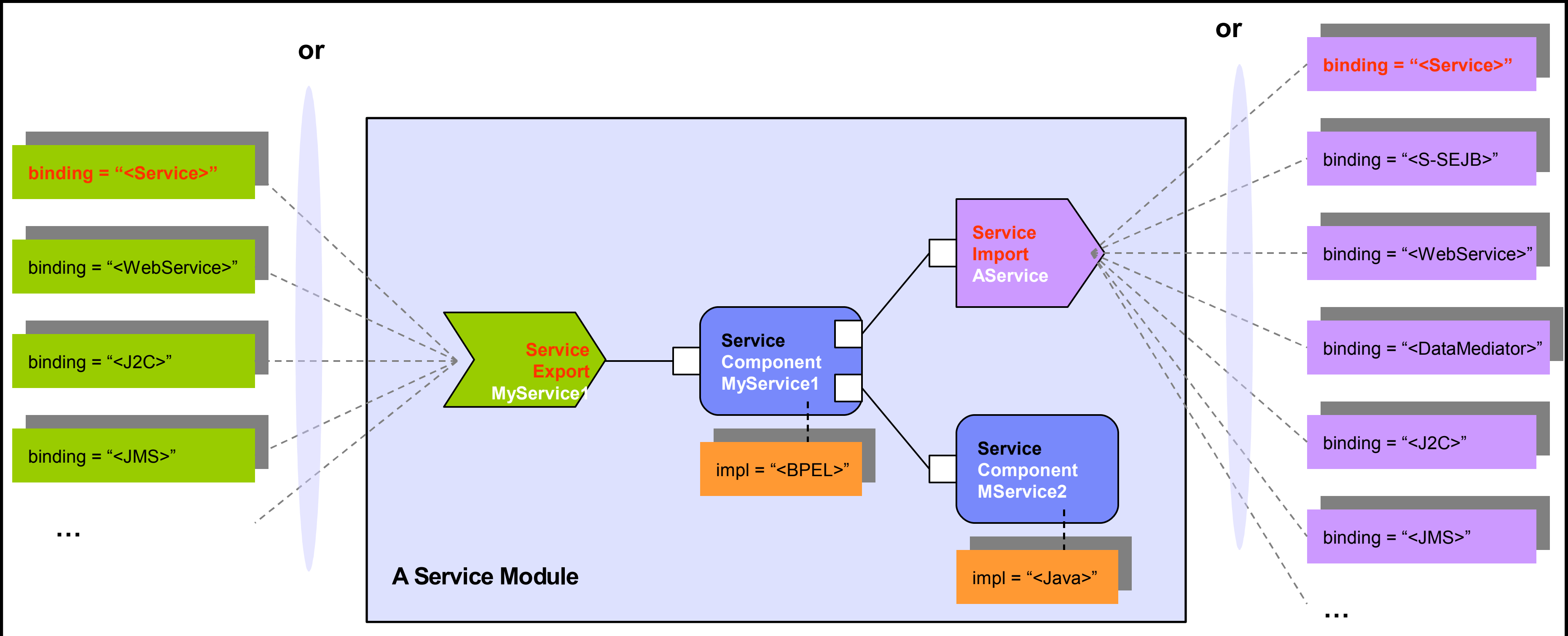
Component Implementation Types

- A Service Component can have any one of a number of implementation types and languages
- Plain Old Java Object (POJO)
  - EJB Stateless SessionBean – including EJBs derived from POJO with annotations
  - Business Process Execution Language (BPEL) Process
  - Adaptive Business object
  - XSLT transformations
  - PHP and Javascript
  - Mediations
  - Portlet
  - Business Rules sets
  - CICS or IMS Transaction program sets
  - SQL query sets
  - RDB Stored Procedures
  - … it's extensible



Inter-Module Dependencies

- Service ports can be exported from a module with an Export
- Service dependencies can be declared in a module with an Import
  - Imports and Exports can be bound to other external services (derived from SCA or non-SCA component abstractions)

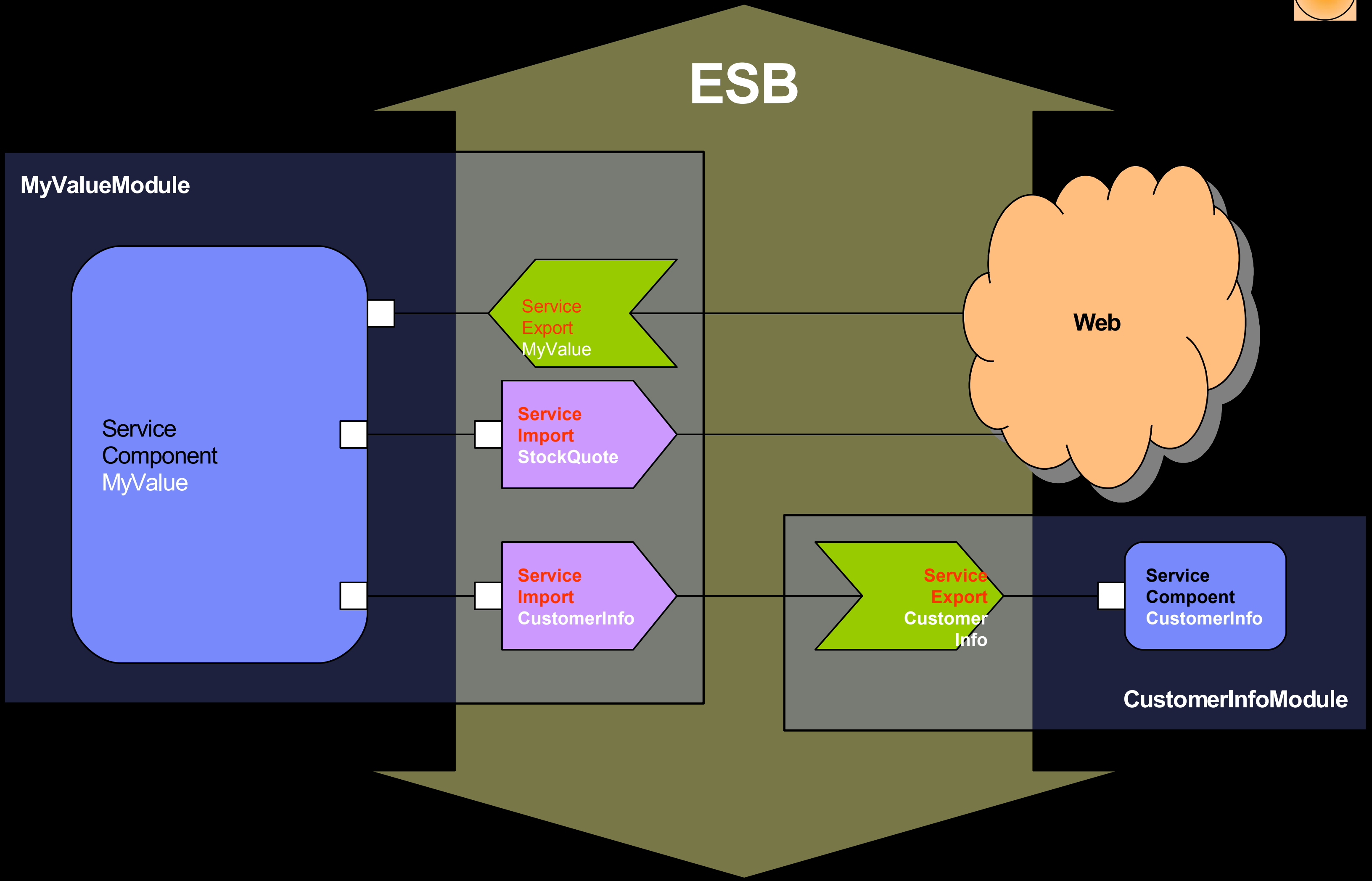


Design
(Models, Patterns, Templates, Policy)

SCA and ESB

- Inter-module services are wired via the ESB at deployment time

Invocation

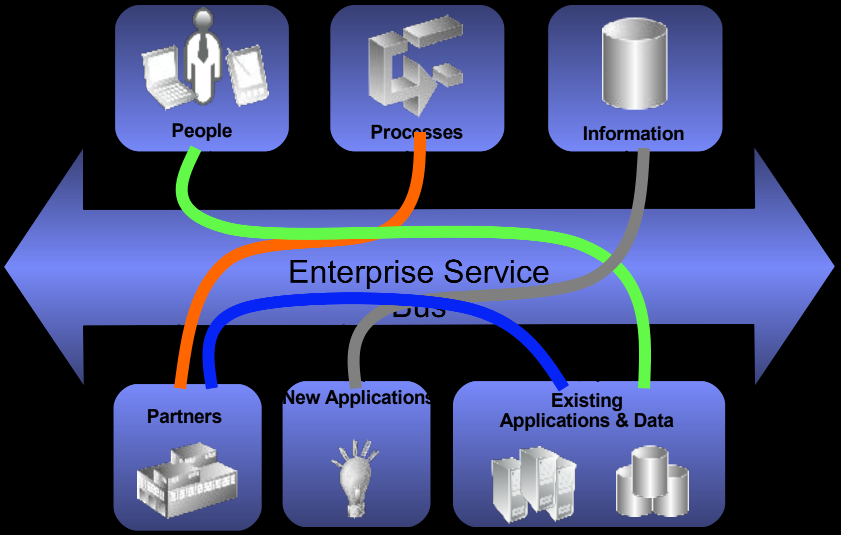


Design
(Models, Patterns, Templates, Policy)

Web Services and SOA

Invocation

- SCA is a component model for services
- Web services provide standard interoperability between services across heterogeneous systems
  - Web services is one of several protocol, encoding and interoperation specifications supported by the ESB
- Remember, SCA component interfaces can be described using WSDL Port-types



Summary:

- We have a complex programming model today, with many competing and/or overlapping technologies
- Service Component Architecture and Service Data Objects give us a simpified, unified programming model that allows us to focus on the business problems rather than the implementation technology, when developing applications for an SOA.
- IBM's Rational and WebSphere tools support SOA development today, including SCA and SDO.