



IBM SOA PoT

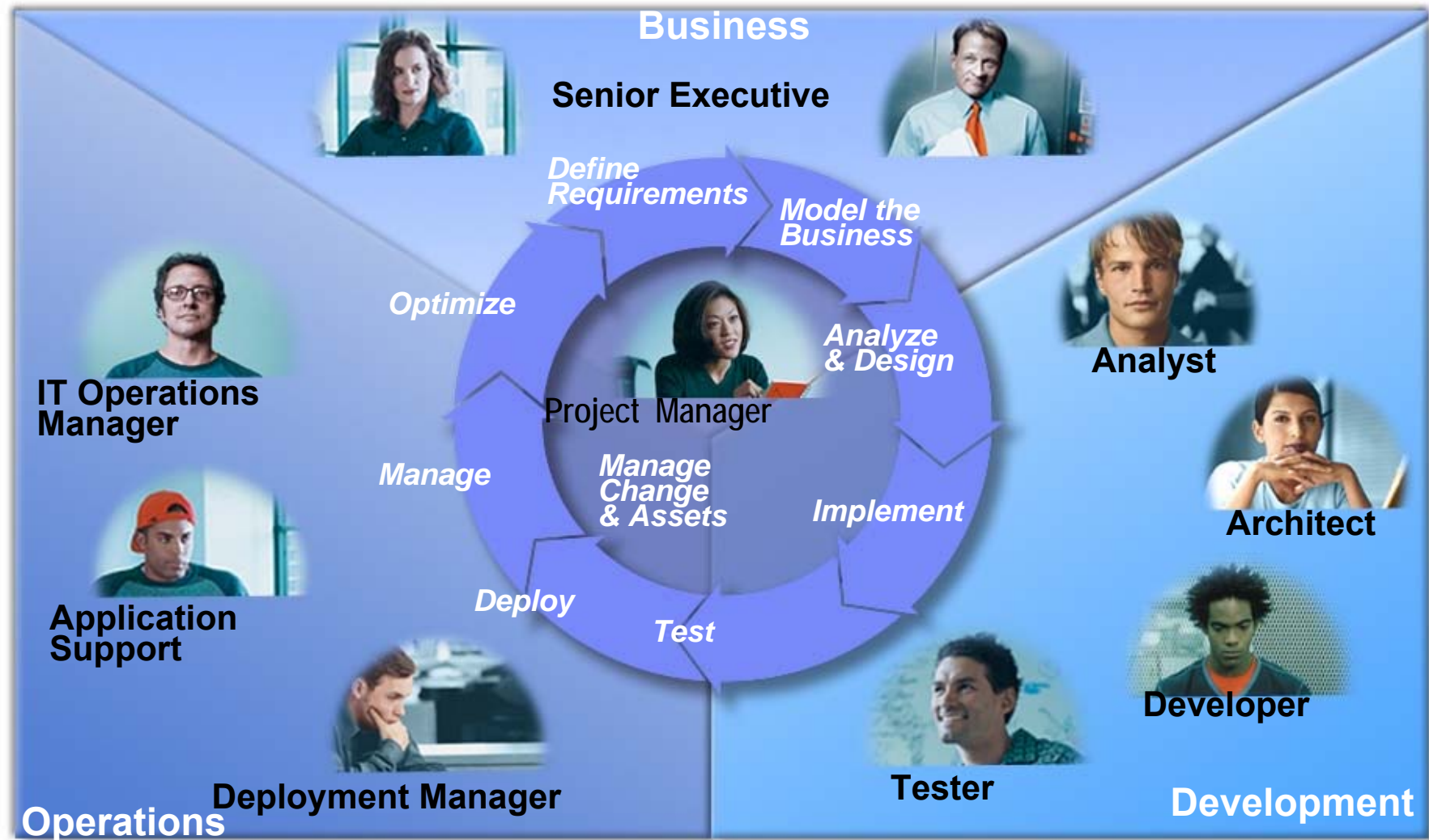
## C3020 Developing New Services








10 July, 2008

SOA Proof of Technology

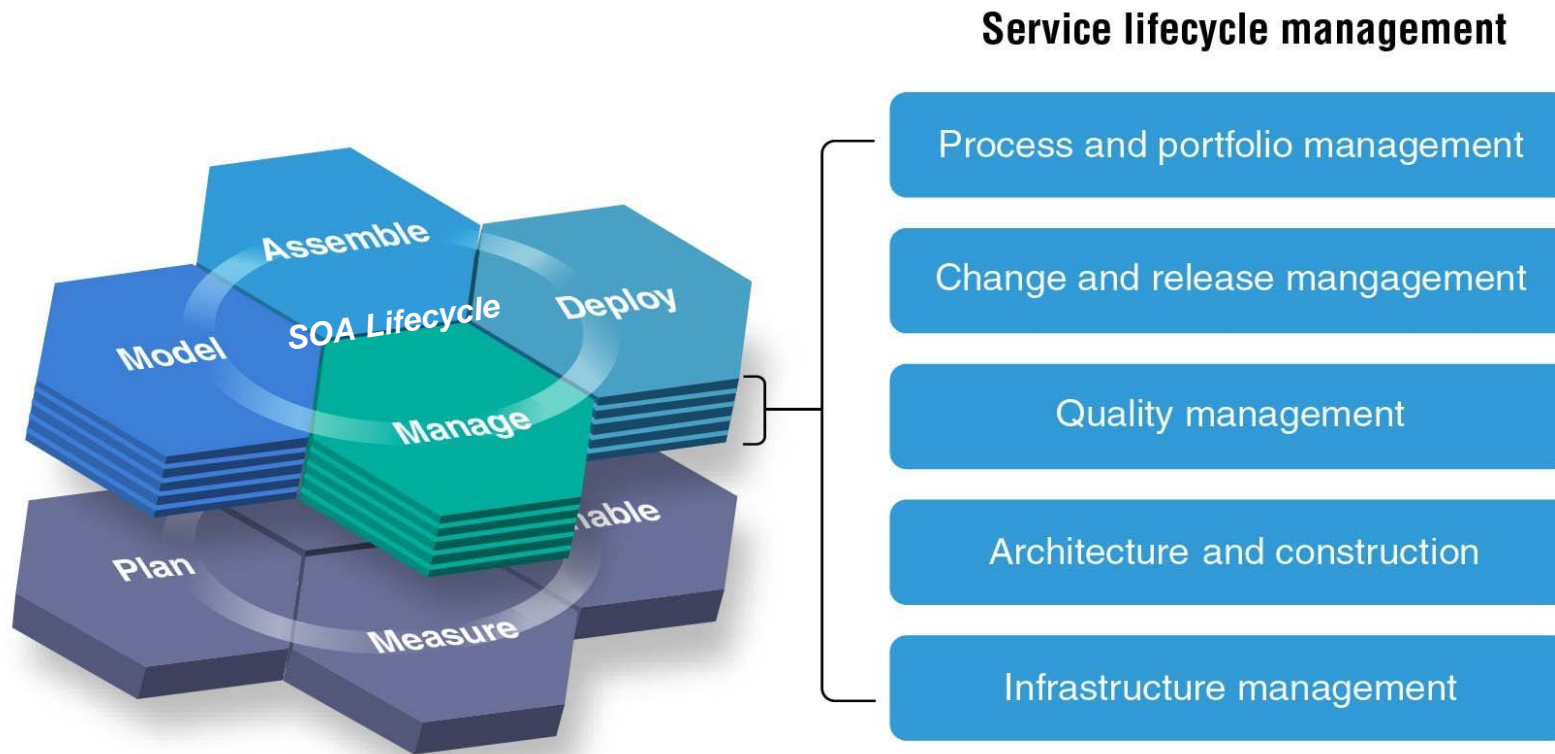
# The business-driven development lifecycle approach



## Business Driven Service Lifecycle Management is a Team Sport

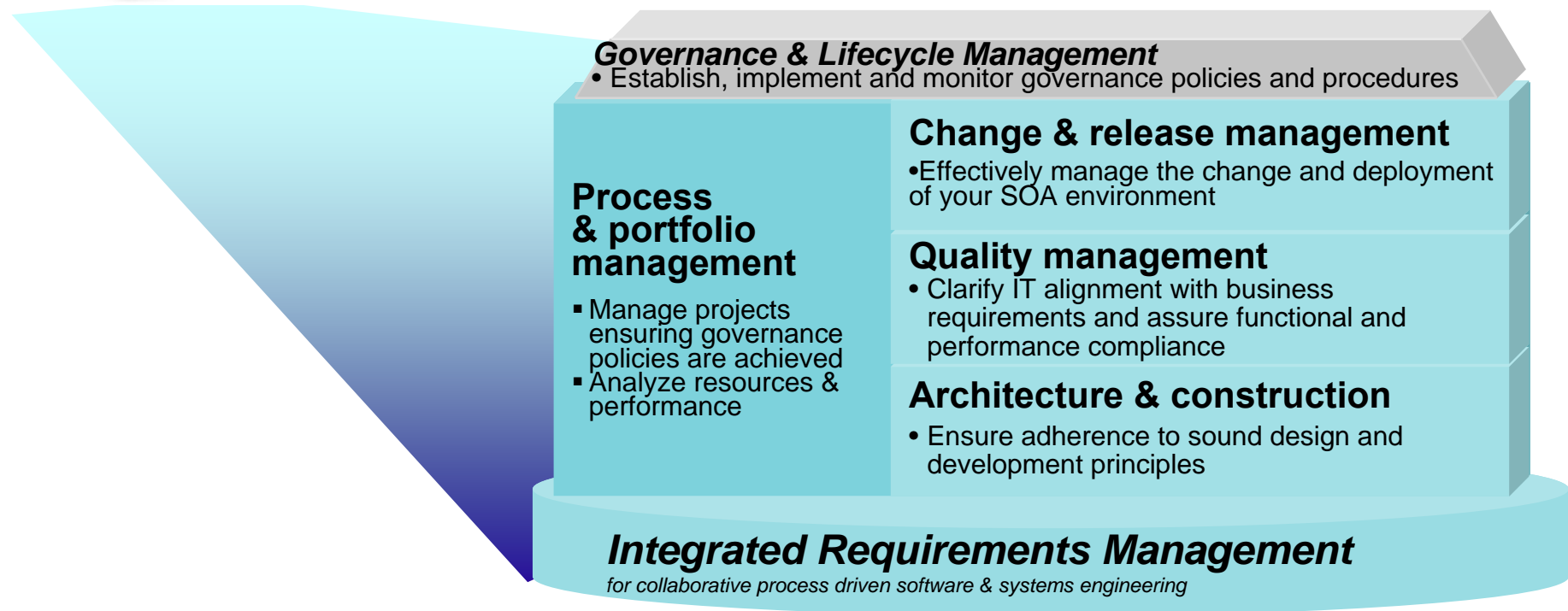
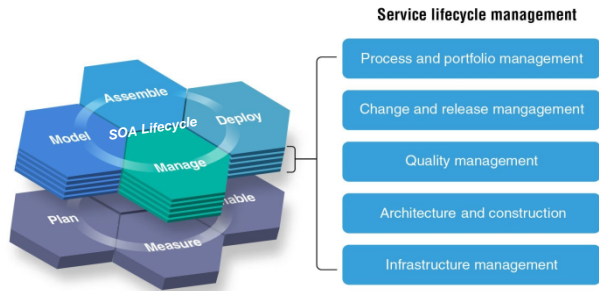
	Title	Role
	<b>Business Service Analyst</b>	Provide comprehensive and structured view of the business requirements and processes
	<b>SOA Architect Service Designer</b>	Provide architecture and design for a software application or service
	<b>Service Assembler</b>	Creation of new business function by combining existing services
	<b>Service Developer</b>	Develop the business services according to the Architects' design, incorporating both functional as well as presentation aspects of the offering
	<b>CIO / Project Manager</b>	Gain visibility into the business benefits, costs and risks of the portfolio of SOA services from proposal to retirement

## SOA Service Lifecycle Management builds upon SOA governance



*Service Lifecycle Management aspects cut across the entire SOA lifecycle*

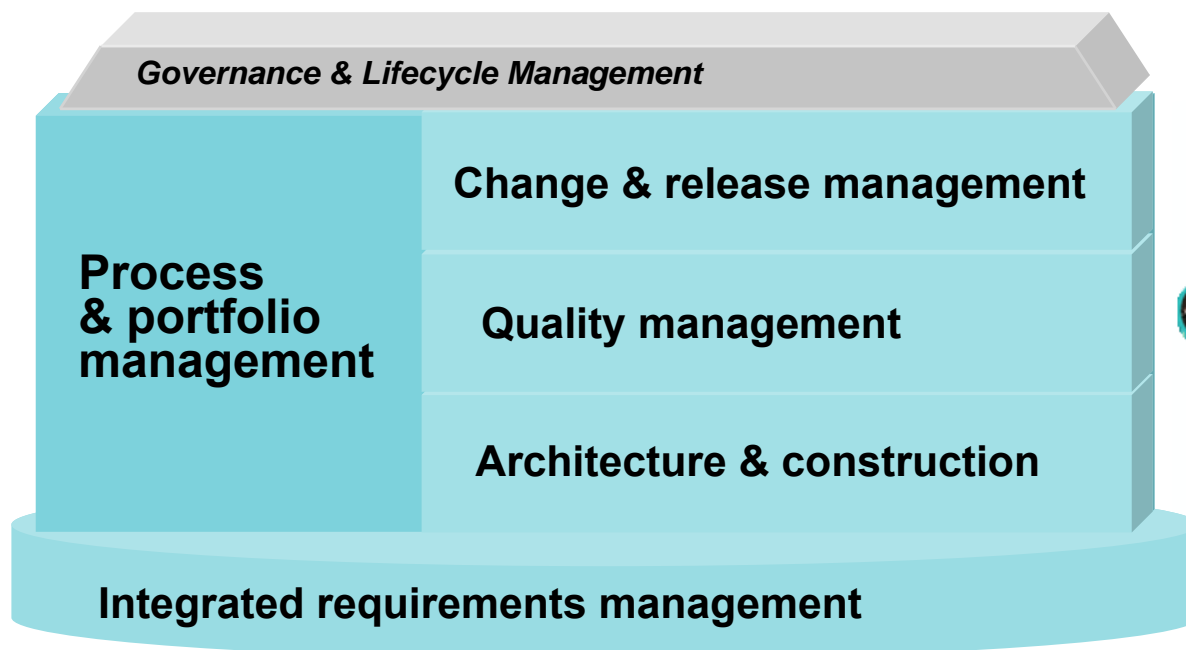
# Service Development & Delivery Defines the Proper Execution of SOA Governance Policies



## IBM Rational Software Delivery Platform - What we offer

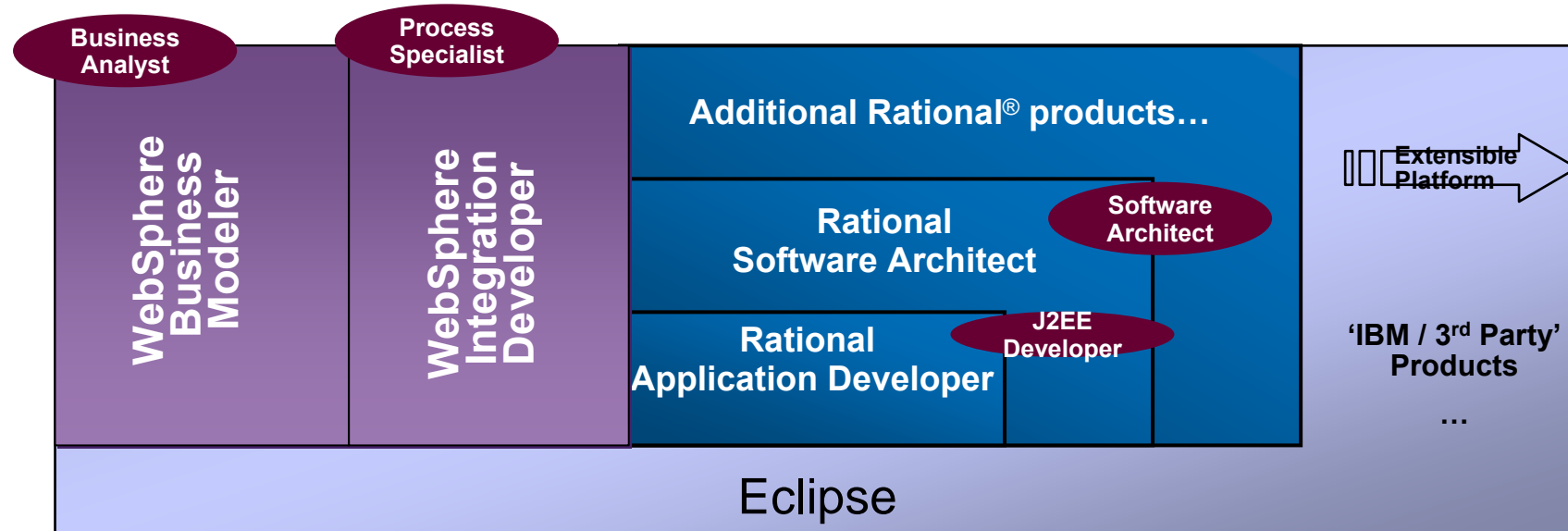
**Rational.** software

*Process design and implementation best practices based on a market leading solutions delivery platform*



- In-house
- Outsourced
- Packaged

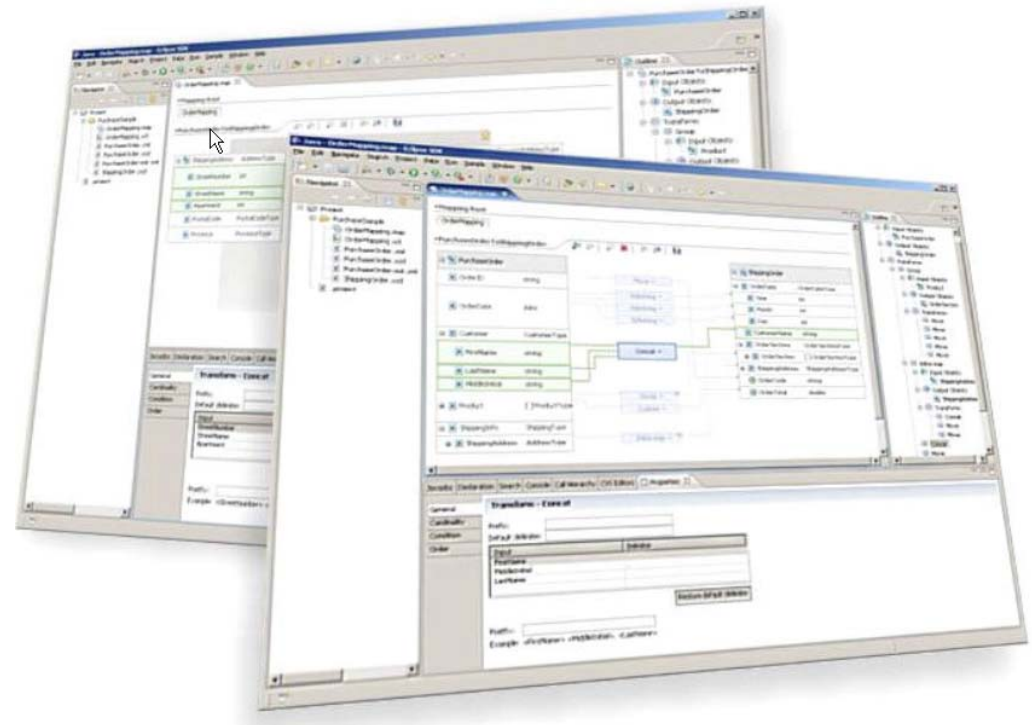
## Tooling Portfolio - Extensible tooling platform built on Eclipse



- **All IBM Software Development platform products install in a consistent and extendable way within the 'platform'**
  - First Product installs the 'platform' as well as it's own product-specific 'installable units'
- **Role-based tool approach, experienced as single integrated "desktop" IDE**
- **Reuse of RAD and Eclipse components ('installable units') is baked into the design**

## IBM Rational Application Developer v7.0

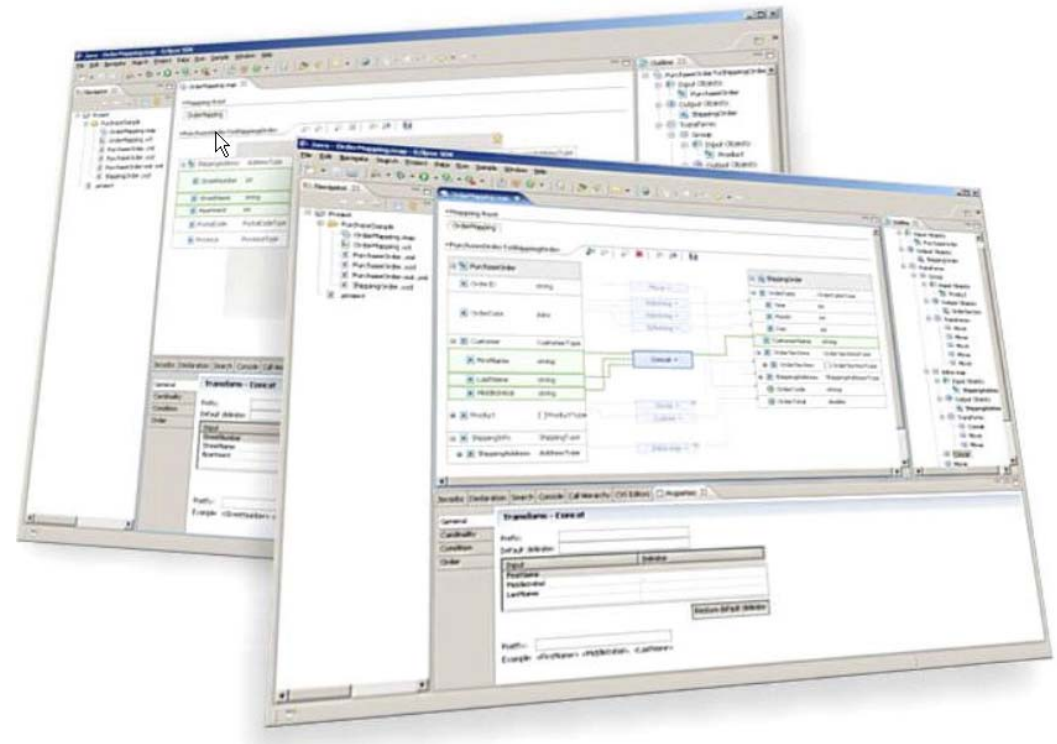
- **IBM Rational Application Developer is an integrated development environment for Java/J2EE and related environments**
- **Accelerate development using rapid application development tools and visual development techniques**
- **Support for extending the development environment and granular installation options**
- **Shorten the Java learning curve through visual design that updates code, point-and-click database connectivity, visual portal development techniques and more**
- **Improve code-level design and share designs with distributed teams using UML visual modeling tools**





## IBM Rational Application Developer v7.0

- ✓ Integrate your business applications with WS-I compliant Web services and service-oriented architectures
- ✓ Collaborate and share assets across the team using built-in Rational ClearCase LT version control
- ✓ Improve code quality with automated tools that apply coding standards and perform runtime analysis
- ✓ Integrated and optimized for IBM WebSphere Application Server and IBM WebSphere Portal Server products



## IBM Rational Application Developer – Key Features

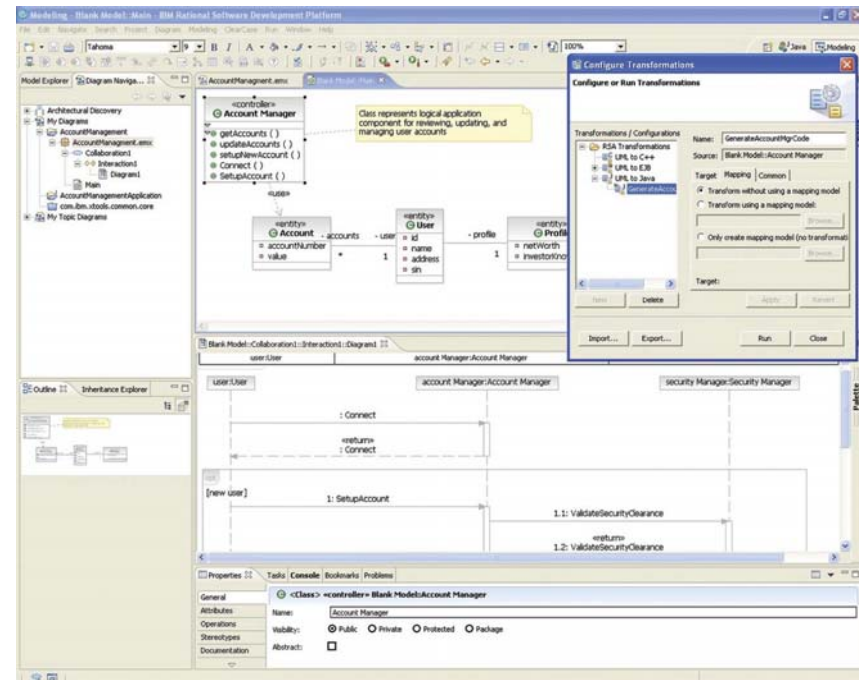
- **Web application development**
  - **Web Site Design, Templates, Styles, Themes and Fragments**
  - **Drag & Drop Page Design and Construction**
  - **Point & Click Database Access**
  - **Event-driven programming model**
- **J2EE application development**
  - **Tools to easily build and test J2EE components with built-in test environments**
- **Rich Client development**
  - **JavaServer Faces, Struts and Faces Client Components for HTML client**
  - **SWING, AWT, SWT support for Java client**
- **Web Services development**
  - **Easily consume and produce Web services from a variety of sources**
- **XML application development**

## IBM Rational Application Developer – Key Features

- **Visual application development**
  - Visualize and graphically edit code and objects
  - UML Visualization of objects for enhanced understanding and productivity
- **Portal & Portlet application development**
  - Visual Portal and Portlet Designer with built-in portal test environment
- **Code quality tools**
  - Automated Code Review
  - Runtime Analysis and Profiling
- **Software Development Process**
  - RUP Configuration for J2EE
- **Web report development**
  - Included Crystal Reports development environment
  - Native Java Reporting Component
  - Embedded Report Designer

## IBM Rational Software Architect v7.0

- IBM Rational Software Architect is an integrated design and development tool.
- Leverages model-driven development with the UML for creating well-architected applications and services.
- Develop applications and Web services efficiently
- Exploit the latest in modeling language technology and leverage an open and extensible modeling platform
- Review and control the structure of Java and service-oriented applications
- Simplifies the design and development tool solution by integrating with other facets of the lifecycle



## What's the difference??

- **RSA is a superset of RAD**
- **RAD is a superset of Eclipse**
- **RSA adds**
  - **UML modeling**
  - **Patterns and code transforms**
  - **Architectural analysis tools**
  - **Reporting (modeling)**
  - **Other stuff**



## Lab Overview

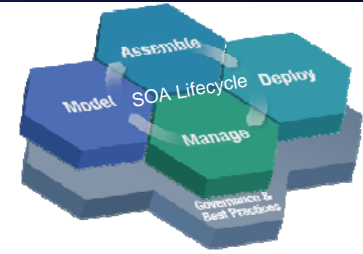
### Service Modeling & Design

In response to the approved activity, the SOA Architect will use the refined business model as a contract to implement a selected task as a service. A design model is developed and stored for reuse at a later time. Using a model transformation, the SOA Architect creates a skeleton implementation of the service in the form of a session bean which is added to version control. A Service Developer is assigned for implementation.

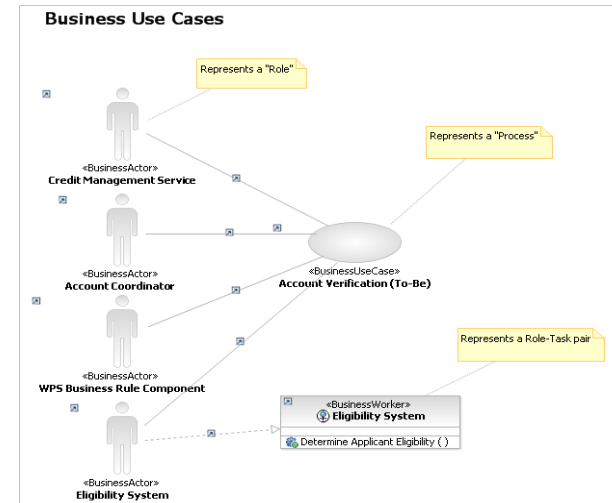
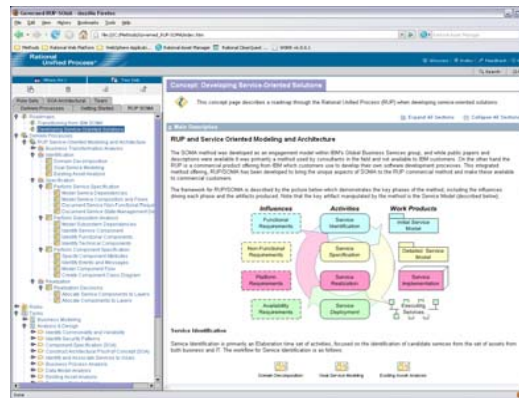
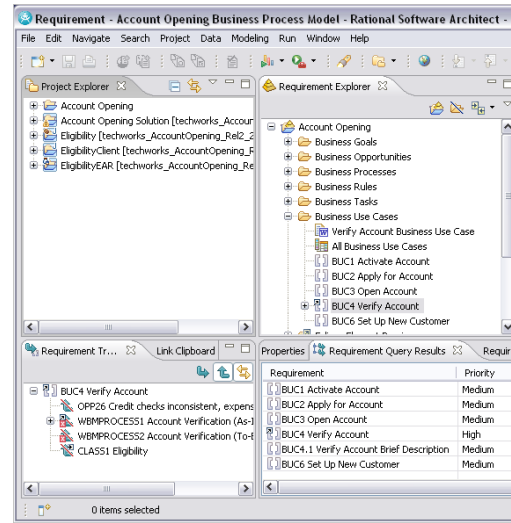
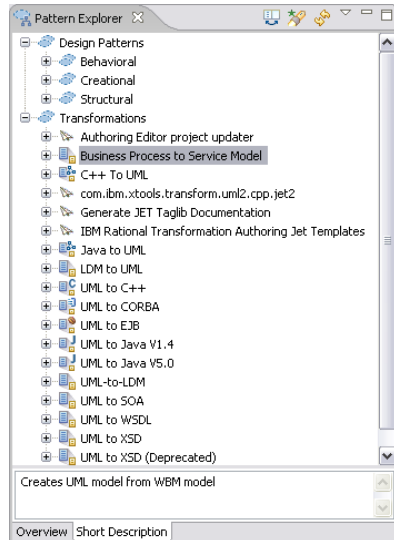
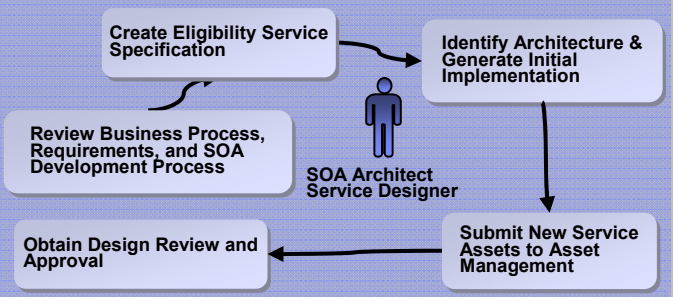
### Service Development

The Service Developer uses the approved activity to initiate implementation of the new service which is imported in a local workspace from version control. The service implementation is accomplished by adding the required business logic and a web service is generated from the implementation. The service is then deployed, unit tested, added to the reusable asset repository, and published to the runtime service registry.

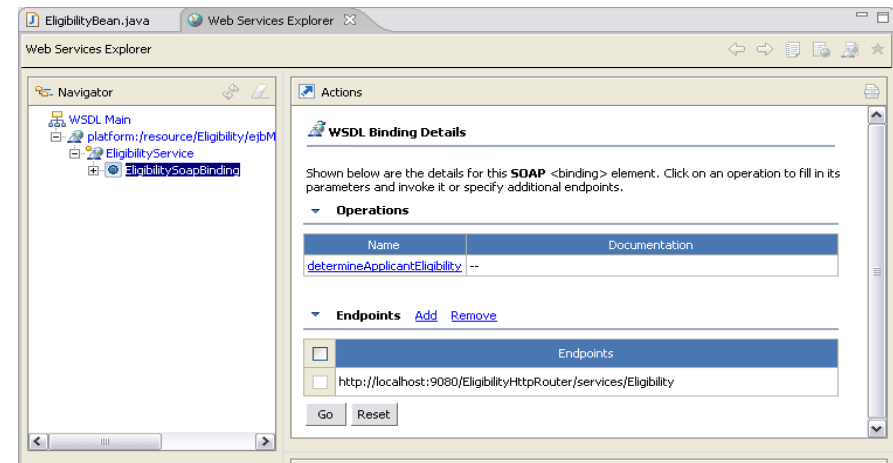
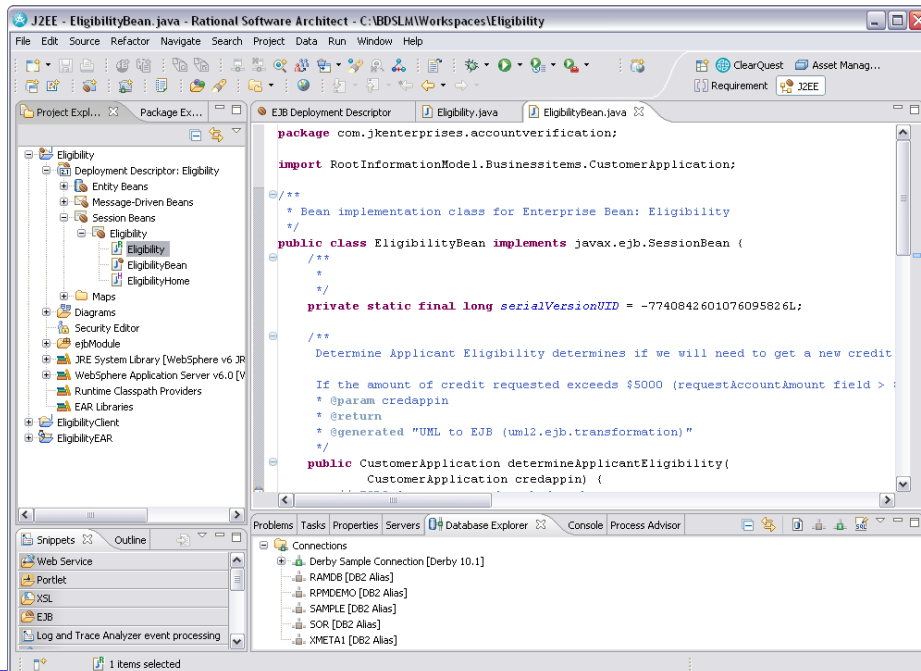
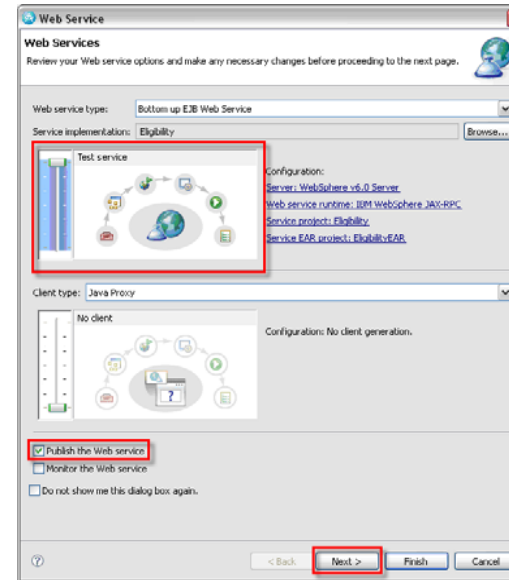
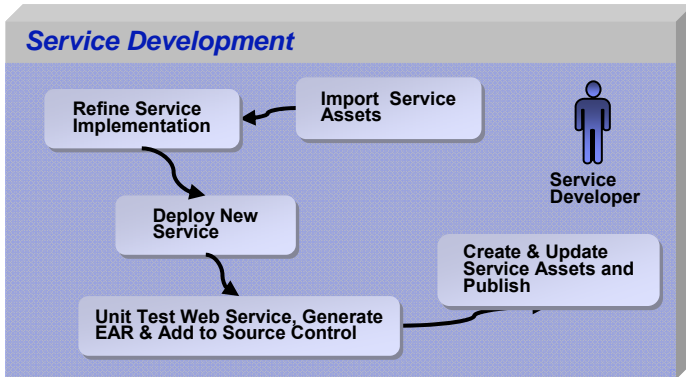
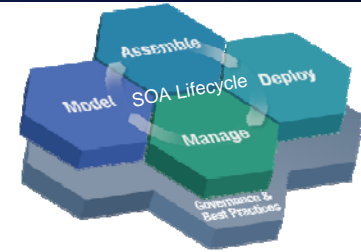
# Service Modeling & Design - Design the Services Architecture



## Service Modeling & Design



# Service Development - Construct the services: assemble & deploy composite application





# LAB 13 - 45 Minutes