An IBM Proof of Technology

SOA (Service Oriented Architecture)

IBM Johannesburg 8th – 10th July 2008

Lab Exercises Part 2

PoT.WebSphere.07.4.031.02

Created for **TechWorks**

Contents

13 Lab   Business Driven Development

Uses VMWare Image 2
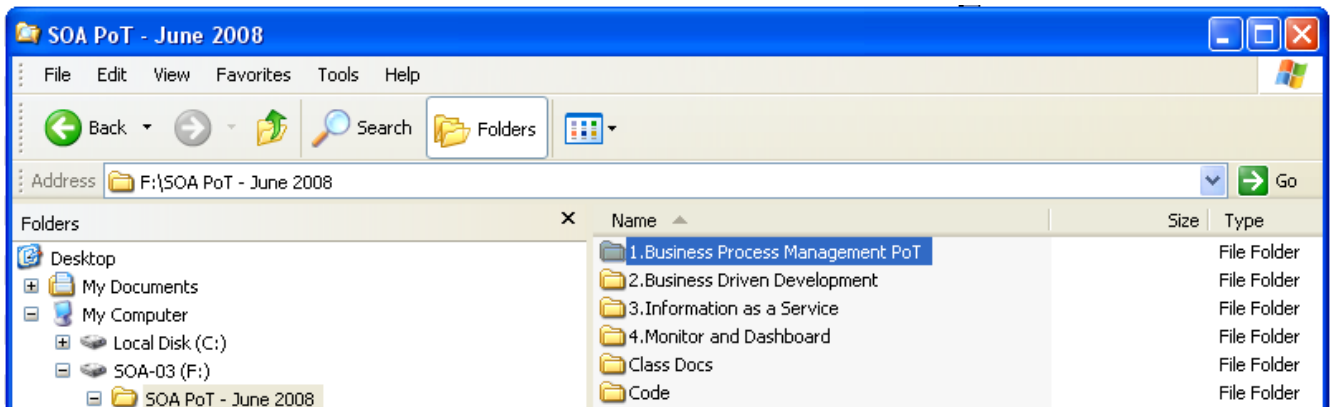
Start the "2.Business Driven Development" image

Lab Introduction

In this module the student plays the role of a Software Architect (Architect). Using a Model Driven Development approach, the Architect uses the business model developed using WebSphere Business Modeler (WBM) as a business contract to implement selected tasks as services. Using IBM Rational Software Architect (RSA), the Architect creates a relationship between the existing business requirements and the model, and specifies new system requirements. Once the candidate task(s) for automation have been identified, the Architect uses patterns and transformations to automatically create the skeleton implementation for the service. Once the service is implemented, the Integration Developer will be able to use it in WebSphere Integration Developer (WID).

Module Overview

The module is divided into four parts. These parts correspond to a set of activities that the Architect performs in a typical Model Driven Development approach. Fig. 1 shows what roles are involved and what tools are used in each part of the module.

___ a.   **2.1**: Review WBM Process Model

___ b.   **2.2**: Browse Requirements and Review RUP/SOMA Process

___ c.   **2.3**: Identify Architectural Elements

___ d.   **2.4**: Generate Initial Implementation

Important: *If you encounter problems during the course of this exercise, please contact the class instructor or any of the module assistants.*

Note: *This module is intended to be completed in **30 minutes**.*
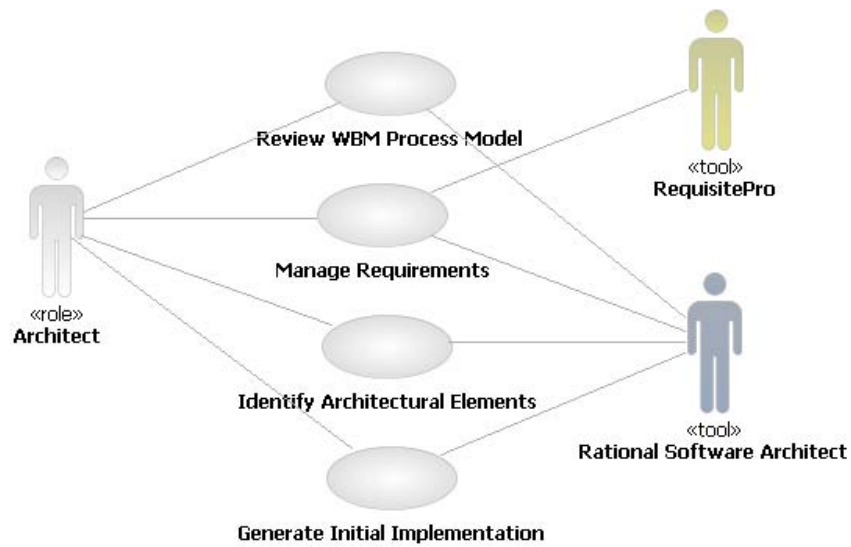

Fig.1 - Roles and Tools used in each part of the module, represented as a Use Case diagram.

(a)

Review the WBM Process Model

You can open WebSphere Business Modeler projects from within RSA. WBM elements are automatically translated into UML elements when the model is opened. You can browse WBM elements and use them to create UML business use case and analysis diagrams. RSA treats the WBM model as a read-only "business contract". Any change in the business model should be done by a Business Analyst using WebSphere Business Modeler. You can see the changes in RSA by re-opening the model.

Open the WBM Process Model in RSA

___1.   Start Rational Software Architect as the Architect

      ___a. Double-click the **Rational Software Architect** shortcut on your desktop

      ___b. Select **Account Opening** by single left-clicking to see the project properties.
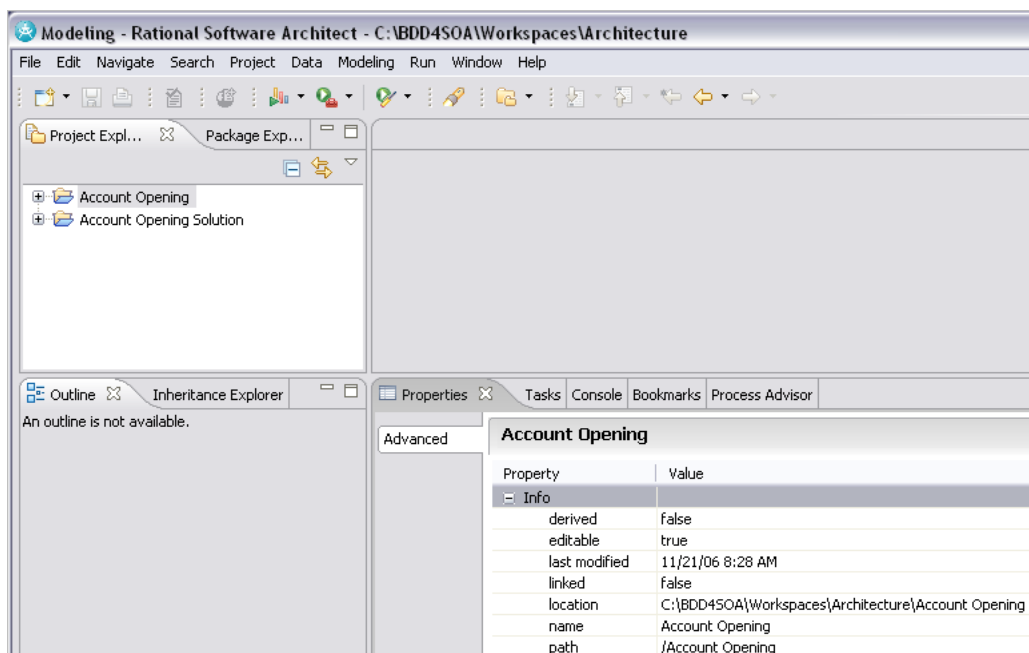
      ___c. Expand the **Account Opening** folder

          The Project Explorer view shows the contents of the WBM workspace.
          Remember that this is NOT an import of the WBM project in RSA. The **Account Opening** project in RSA is only a reference to the WBM project.

> ⚠ **Caution!**
> If you delete the Account Opening project in RSA, leave the "Do not delete contents" default selection, otherwise you will delete your WBM project.

___d. Double-click the **Account Opening** file (shown above)
Wait until the translation is finished. The name changes once RSA has read the business process model.
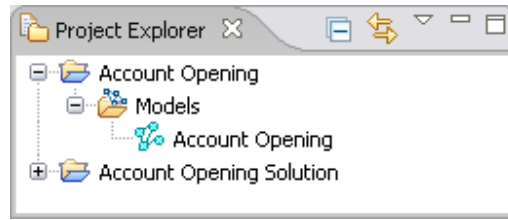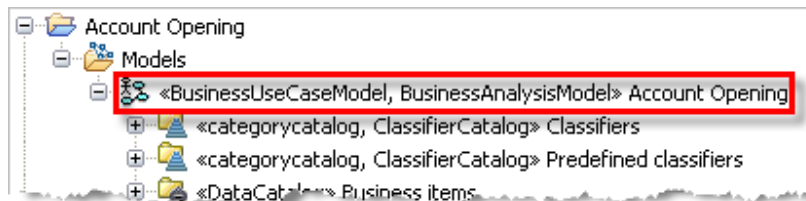

> ℹ️ *When you open the resources.XMI file, a transformation automatically converts the WBM elements into UML elements.*

Browse the WBM Elements

___2. Expand the model hierarchy

    ___a. Navigate to:
Account Opening ->
Models ->
<<BusinessUseCaseModel, BusinessAnalysisModel>> Account Opening ->
<<ProcessCatalog>> Processes ->
<<ProcessCatalog>> To-Be Processes
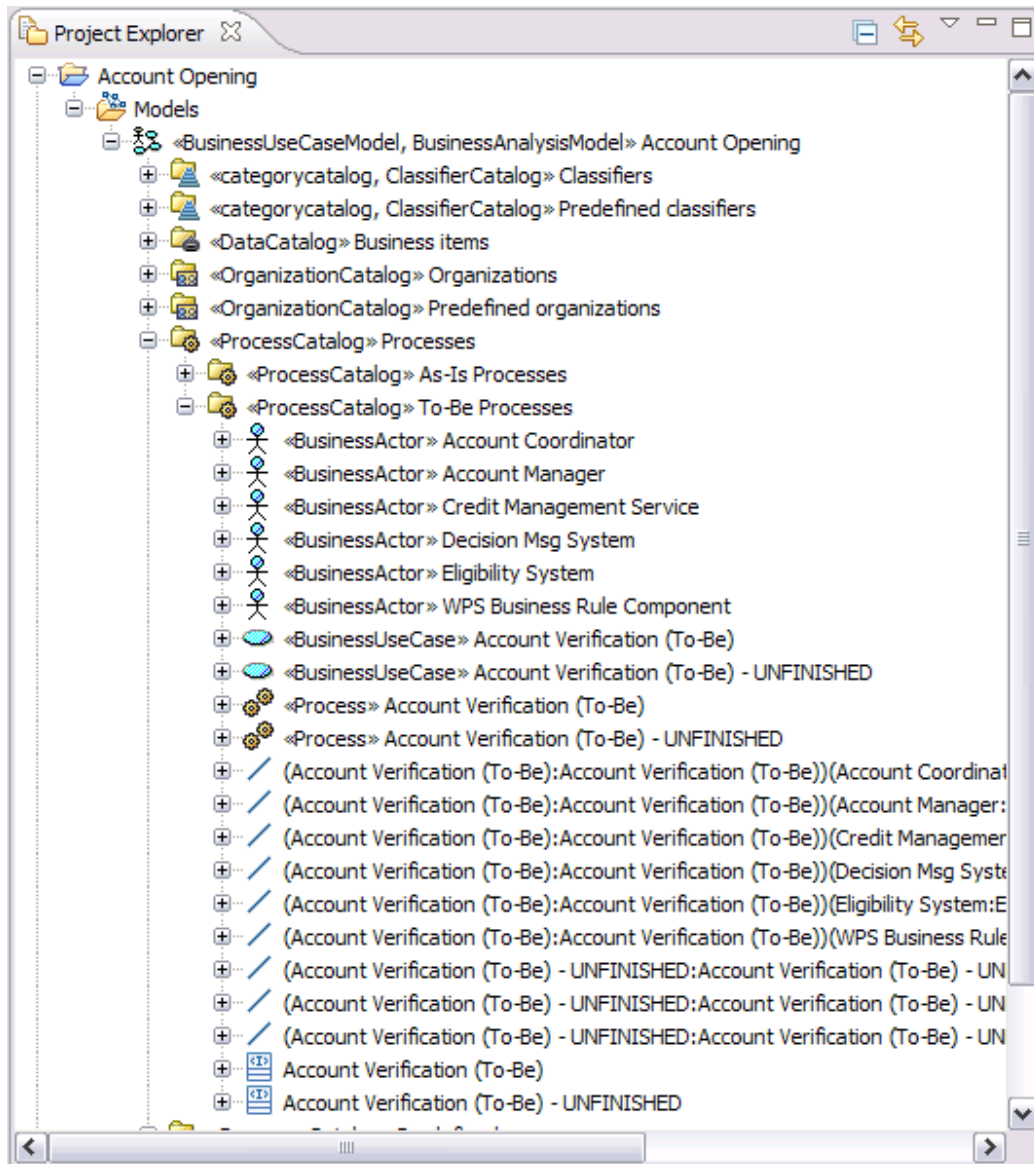
> ℹ️ *You can see that all the elements in WBM were translated into UML elements: folders into packages, roles into business actors, processes into business use cases, etc.*

___3.   Get familiar with the UML element types

          ___a. Select the **Eligibility System** (Actor)

          ___b. Select the **Account Verification** (Use Case)

          ___c. Select the **Account Verification** (Process)

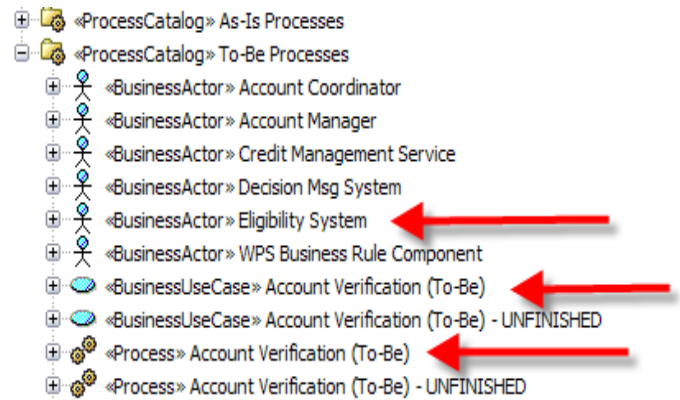> *Each time you select one element in the "Project Explorer" view, the **Properties** view will show you the type of the element selected (in our example, respectively: Actor, Use Case, Collaboration). This is a very easy way to get familiar will all the UML elements types that are in our Account Opening project.*
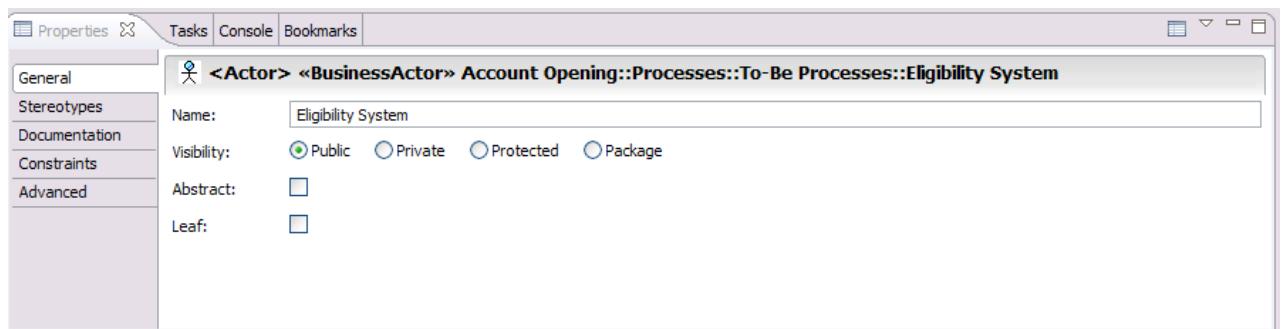
          ___d. In case you don't see the Properties view, go to: **Window -> Show View -> Other... -> Basic -> Properties**

___4.   Browse the "Business Items"

          ___a. Expand the **<<DataCatalog>>Business Items** package.
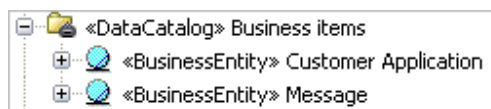
> *Note how all the attributes of the business items have been correctly translated.*
>
> The "Account Opening Project" is read-only. If you want to make changes to the business model, you will have to open it with WebSphere Business Modeler, make the changes, and re-open the WBM model file in RSA.

IBM *TechWorks*

View the UML Project

At this point, the Architect creates architectural elements that will fulfill the "contract" represented by the business model.

The Architect creates a new RSA project to separate the business model (the WBM file) and the system models (where use cases, analysis and design elements will be created), and establishes "external references" between the models.

To make this exercise simpler, we created the UML project and the following UML models and diagrams:

As you can see, the Architect already created the following elements in the Design Model…

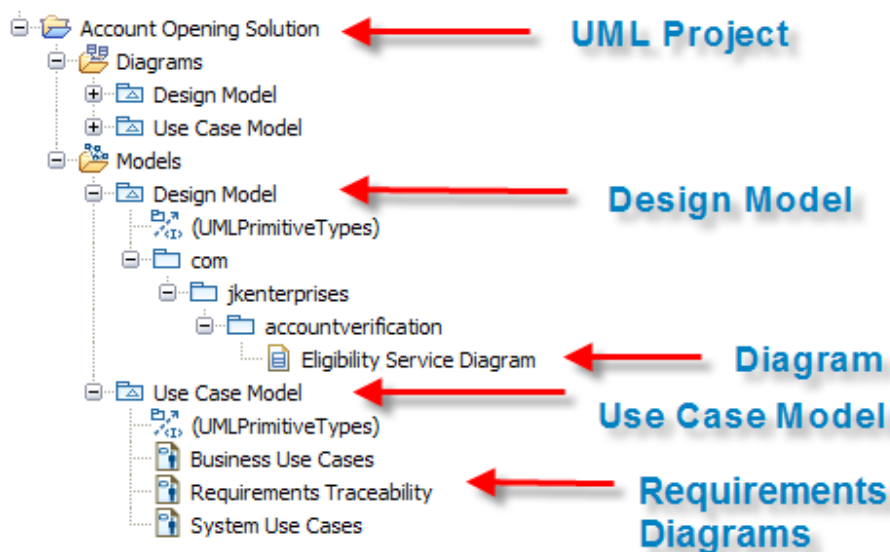| com.jkenterprises.accountverification	packages that will contain our design elements (this way the Java™ code we will generate will have the proper namespace)
Eligibility Service Diagram	the diagram in which we will derive design elements from the business specification

… and the following diagrams in the Use Case Model:

Business Use Cases	to visualize how WBM elements are related
Requirements Traceability	to trace WBM elements to existing and new requirements. We will populate this diagram in the next exercise (Manage Requirements)

Note that in this example we use a single UML Model for Business and System Use Cases. You could create a separate model for the business requirements.

Browse the UML Diagrams

The Architect has already populated several diagrams. Let us start with the "Business Use Cases" diagram. Here's how your "Business Use Cases" diagram should appear:
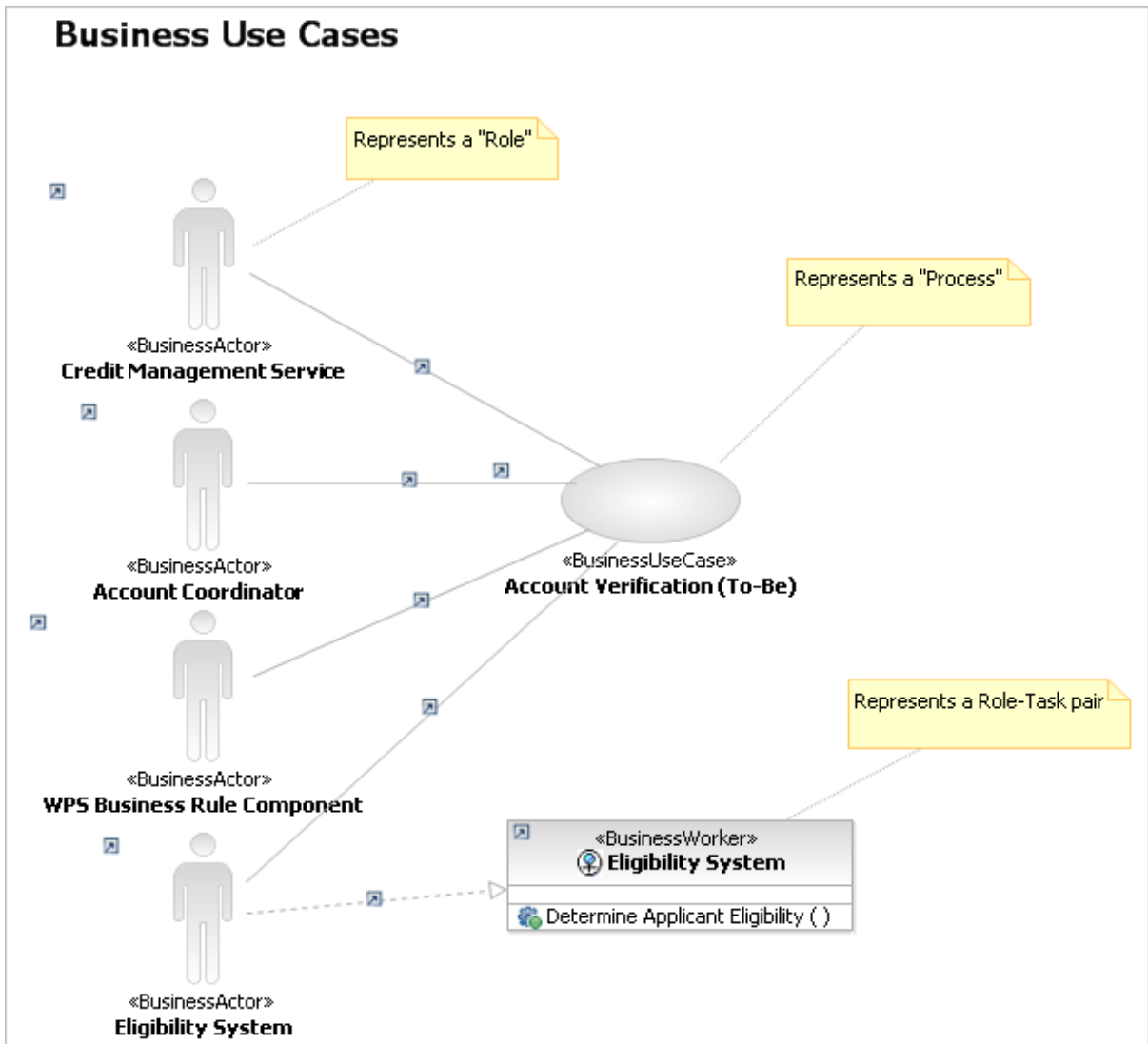
Every UML element in the diagram has a little "arrow" ▧ symbol to show that the element is an <u>external reference</u> to a different model (our diagram is inside the "Use Case Model", its elements were taken from the "Account Opening" model).

If you right-click any element in the diagram and select Navigate ➜ Show in ➜ Project Explorer, the Project Explorer will highlight the location of the selected element.

This is how the WBM ➜ RSA transformation maps the elements:

| WBM Element	UML Element
"Account Verification" Process	"Account Verification" Business Use Case
"Eligibility System" role	"Eligibility System" Business Actor
"Eligibility System" role associated to the "Determine Applicant Eligibility" task	"Eligibility System" Business Worker (interface) with a "Determine Applicant Eligibility" operation

Browse Requirements and Review RUP/SOMA Process

Some business goals (high-level requirements that explain "why" we are interested in modeling and automating certain business processes) have been previously saved by the Business Analyst in RequisitePro.

The Architect populates a UML diagram to visually trace business goals to the business use cases.

___i.   Open the RequisitePro project

___a. Open the Requirement perspective in RSA by click the Open Perspective button then selecting **Requirement**.
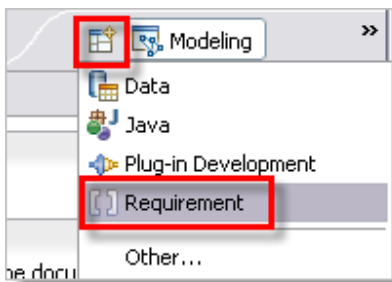
___b. In the "Requirement Explorer" view, double-click the **Account Opening** folder to open the project
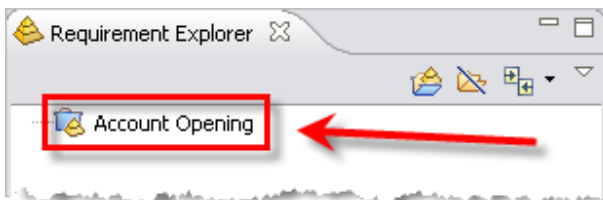
___c. Click **OK** to confirm the "**Unknown**" username (we did not setup security for this project)

RequisitePro can organize several types of requirements (you can create any type of requirement and add any number of attributes to each). In this project the Business Analyst decided to keep track of the following types of requirements:

___1. Business Goals

___2. Business Opportunities

___3. Business Rules

___4. Business Use Cases

___5. Glossary Entries

___6. Supplementary Requirements

___7. System Features and Vision

___8. System Use Cases
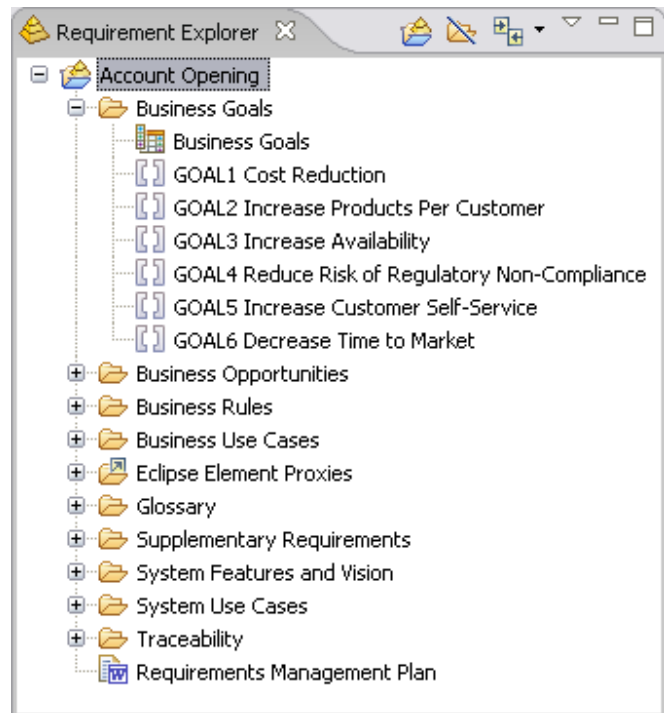
___ii.   View the Properties of the Business Goals (ROI, Cost, Benefit, Priority, etc.)

___a. Double-click on the "**Business Goals**" query (shown above) and see the view query results.
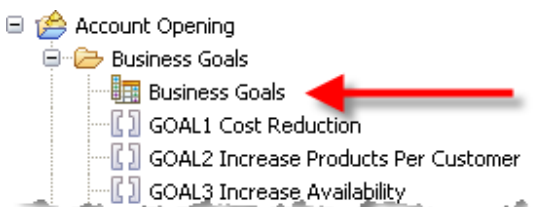
___b. Return to the **Modeling** perspective.

___iii.   View Process Advisor guidance

___a. In the Model Explorer view, double-click **Account Opening Solution** or use the + symbol to expand the view.

___b. Expand the **Account Opening Solution -> Diagrams -> Use Case Model** package by double-clicking

___c. Click on the **Process Advisor** view to review its contents.

___d. Click on the **Requirements Traceability** in the project explorer

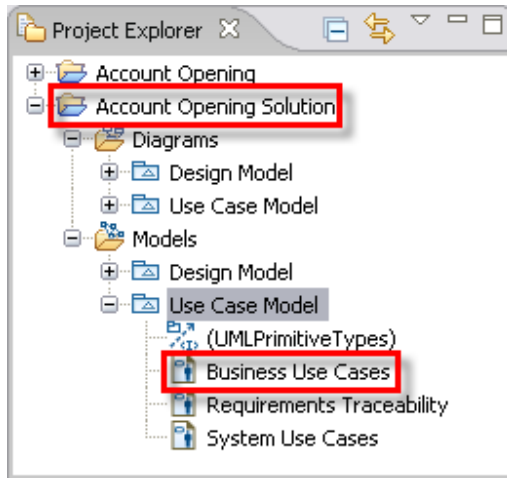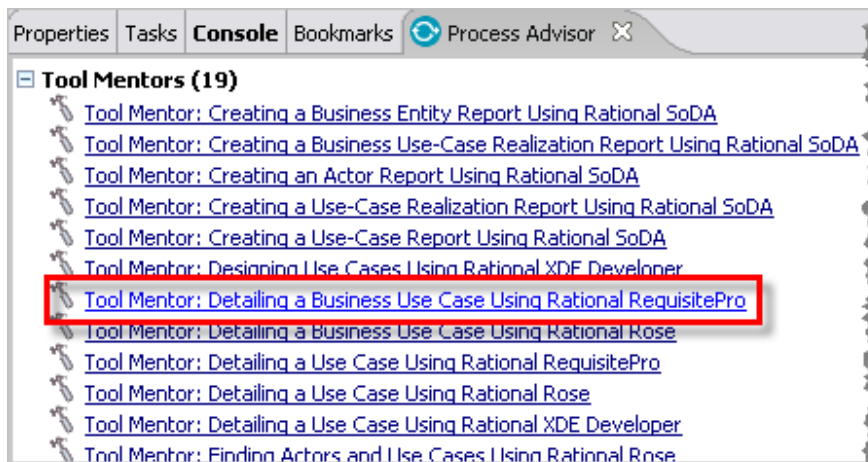> *The Process Advisor view provides context sensitive guidance throughout the Rational Software Architect product experience. Process Advisor provides step-by-step tool instructions through tool mentors, links to artifacts related to the current task, task guidance, as well as general activity guidance.*
>
> The Process Browser, which the next few steps will review, provides full access to the Rational Unified Process built through the Rational Method Composer. The process being reviewed here has Service-Oriented Modeling and Architecture elements blended into the overall process.

___iv.   Review the Process Browser

   ___a. Select **Help -> Process Browser** from the menu
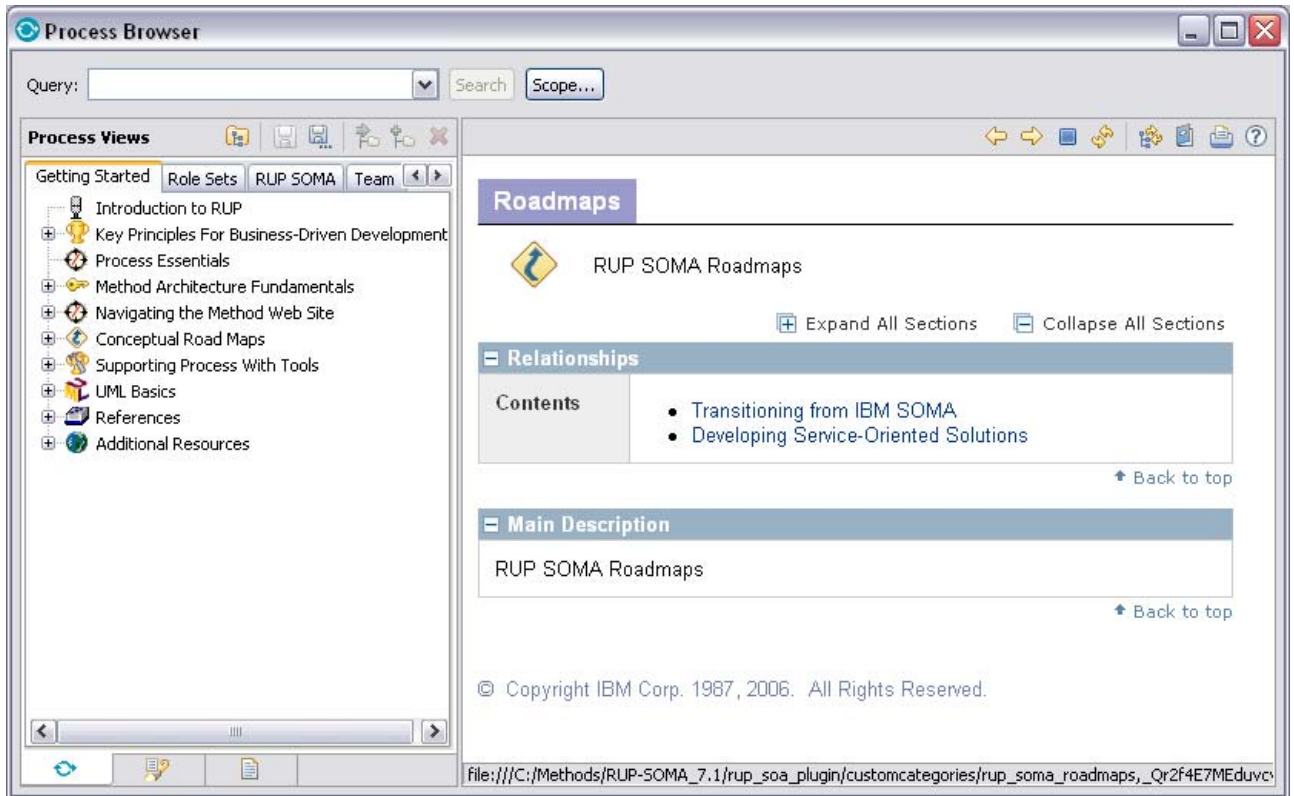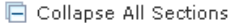
   ___b.   Click on the **Developing Service-Oriented Solutions** link to review detailed
          guidance on how to develop service-oriented solutions using the Rational Unified
          Process Service-Oriented Modeling and Architect method.

___c. **Exit** the Process Browser by clicking on the **X** in the upper right.

___**d.** Close the Use Case Model::Business Use Cases view by clicking on the **X**

Identify Architectural Elements

The Architect uses WBM elements not only to trace and organize requirements, but also as a starting point to model the design and the data of the application.

In the "Account Eligibility" process, the Architect identifies the "Determine Applicant Eligibility" task as a candidate for implementation as a service.

Identify Services

The Architect has already organized the Design Model into packages. We will define the design of the service using the "Eligibility Service" class diagram.

___i.   Open the Design Model

   ___e. Double-click on **Account Opening Solution -> Models -> Design Model** to expand the package

   ___**f.** Browse down to the **Eligibility Service Diagram**

   ___g. Double-click the **Eligibility Service Diagram** to open it. The diagram is empty.
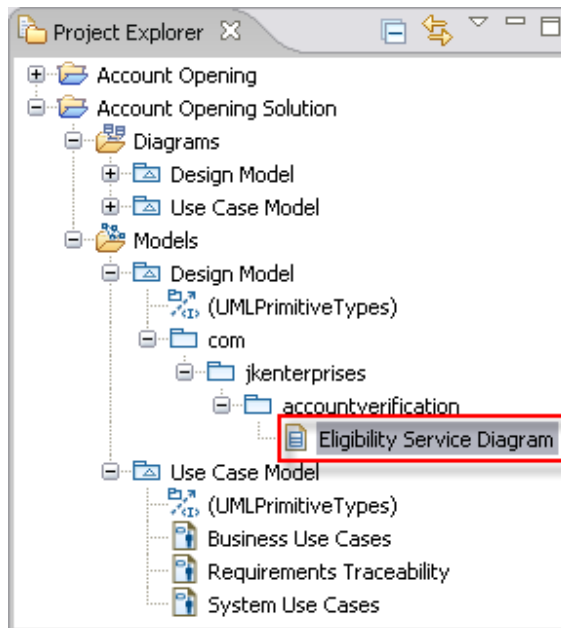
___v.   Identify the interface that will be the starting point for our implementation.

   ___a. Using the **+** sign, expand the **Account Opening Solution -> Models -> Use Case Model** branch to expose its contents.
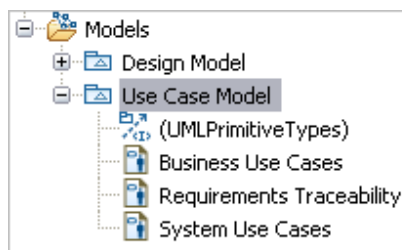
___b. Double-click on the **Business Use Cases** diagram you browsed before
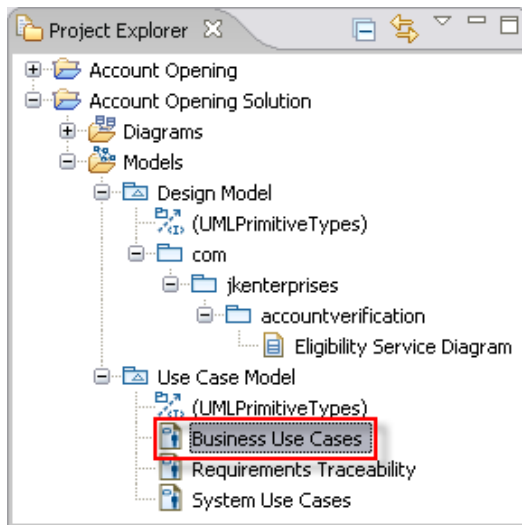
___**c.** Right-click on the **Eligibility System** business worker and select **Navigate -> Show in -> Project Explorer**

This will be the starting point for the design of our service.
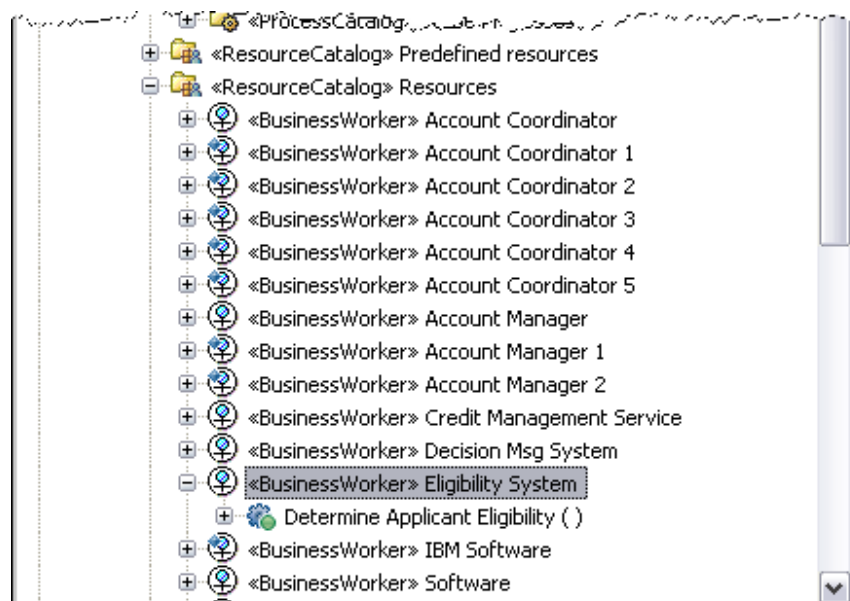
___d. Return to the **Eligibility Service Diagram** by clicking on its tab.

Design Model::com::jkenterprises::accountverification::Eligibility Service Diag... | Use Case Model

> *Note that the "Eligibility System" belongs to the WBM model.*
>
> The "Eligibility System" is a UML interface stereotyped as a <<Business Worker>> (look in the "Properties" view when the interface is selected).
>
> The interface was automatically created during the WBM→UML translation because our business model has an association between the "Eligibility System" role and the "Determine Applicant Eligibility" task.
>
> That association is interpreted in UML as an interface (the role) that has a responsibility (the task). If the role were associated to more tasks, you would see more operations inside the interface.

___**e.** Drag and drop the **Eligibility System** interface from the Project Explorer view onto the **Eligibility Service Diagram**

___vi. Create a new class that will implement the interface

___**a.** In the "**Palette**" view (usually on the right of your diagram), select **Class**

___b. Click the cursor inside the diagram to **create** the class

___c. Name the class **Eligibility**

___vii.   Add an operation to the new class that matches the business contract. We will Copy & Paste the existing operation from the interface into the new class.

___a. Using the Project Explorer view, copy the **Determine Applicant Eligibility()** operation by right-clicking on the operation then selecting **Copy** from the menu and paste it on top of the new Eligibility class by right-clicking on the new Eligibility class then selecting **Paste**.
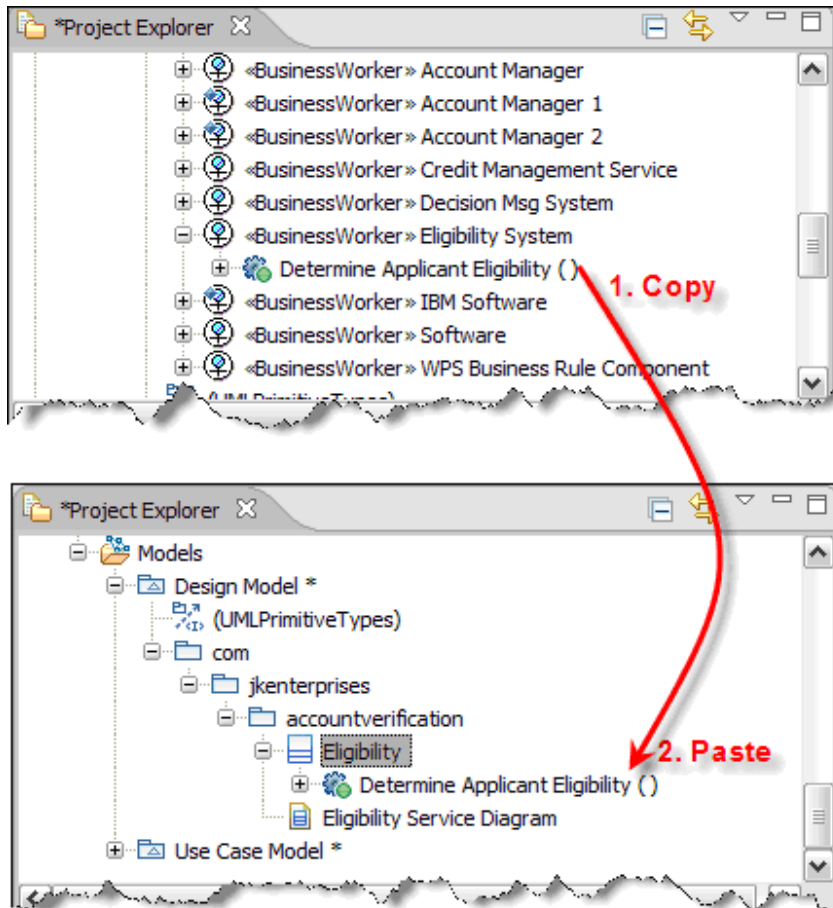
___viii.   Create a "trace" relationship between the service and the business worker

___a. In the **Palette** view, select the **Class** tab then select the **Abstraction relationship** (in the drop-down from the **Dependency** association)

___b. Drag a line from the new **Eligibility** class to the **<<BusinessWorker>>**, select **Create Trace**

___ix.   Apply the «service» stereotype to the Eligibility class

___a. Select the **Eligibility** class
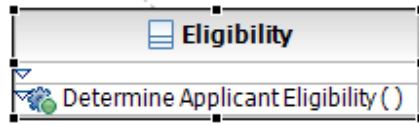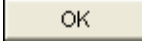
___b. In the "**Properties**" view, select the "**Stereotypes**" tab
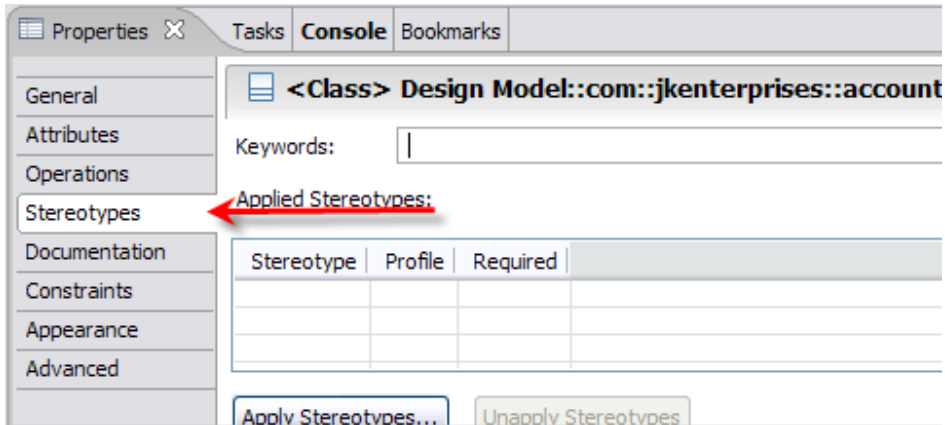
___c. Click **Apply Stereotypes** button
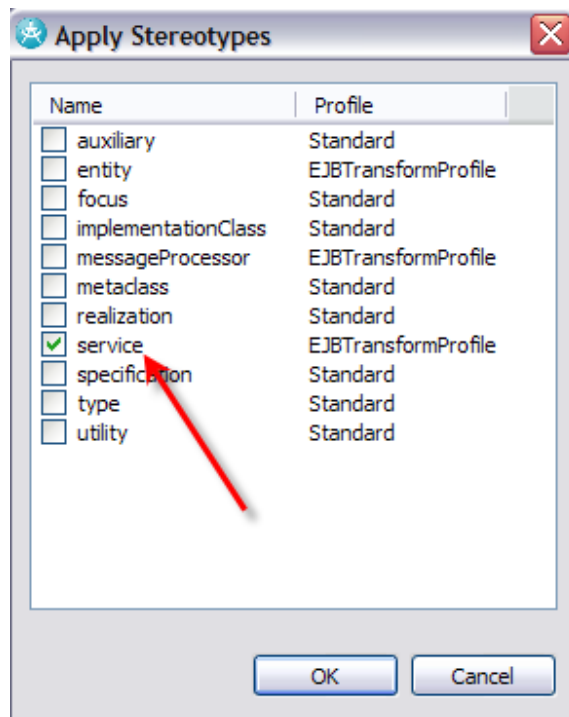
___d. Select the **service** stereotype, click

A stereotype is an UML concept. A stereotype can be added to any UML element (in this module we added the «service» stereotype to the class Eligibility), and gives "extra" properties to the element. A stereotype is also used by UML transformations: the transformation looks for specific stereotypes in a model, converting them into a new UML object or into implementation code, depending on the selected transformation (in our example, we will apply a transformation that recognizes a «service» and converts it into an EJB, generating all the required Java code automatically).

In RSA you can define your own stereotypes and your own transformations.

Here's how your diagram should appear:
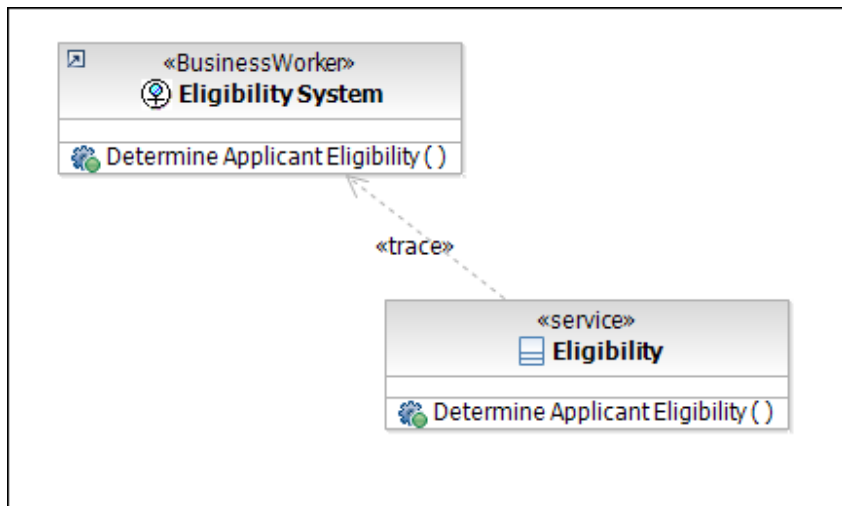
Trace Service to Requirements

Establishing links between the customer's business goals, business use cases, system use cases and the model elements that support them is a key component of traceability. Here, you will trace the new Eligibility service class to the system use case it supports.

___x.   Review use case requirements

    ___a. Switch to the **Requirements** perspective to easily work with requirements

    ___b. Click the **+ sign** to open the **Business Use Case** folder in the Requirement Explorer view
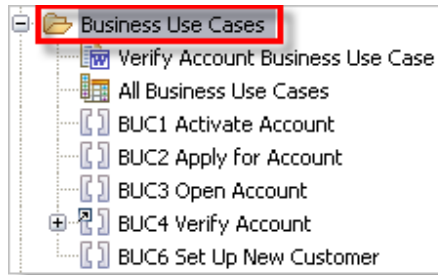
___c. Select the **BUC4 Verify Account** then drag-and-drop it on the **Eligibility** class in the **Design Model** diagram creating a link between the two elements. You may need to scroll around in the class diagram to see the Eligibility class.
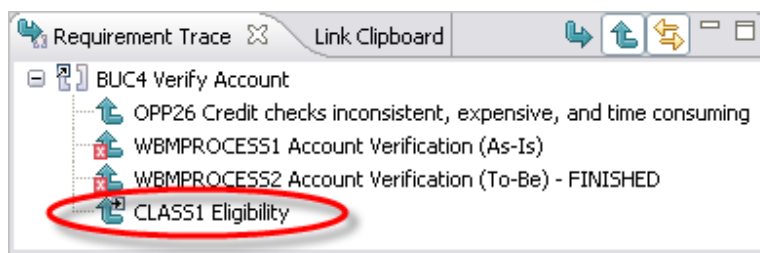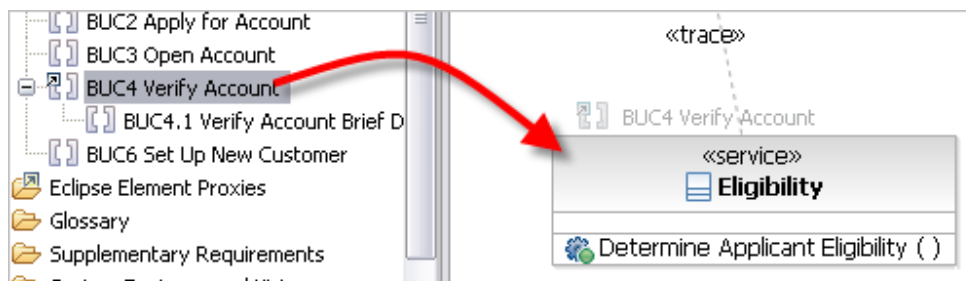
The Requirement Trace view for BUC4 Verify Account now shows the traceability created in WBM to the Account Verification (As-Is) and the new relationship to the Eligibility class created to implement the service.

___d. Switch back to the **Modeling** perspective

Model Data

The Architect looks at the parameters of the "Determine Applicant Eligibility" operations and drags the "Customer Application" business entity in the diagram.

___xi.   Visualize the operation signature of the interface

___a. Right-click on the "**Eligibility**" service, select **Filters ➜ Show Signature**. Move the Eligibility class around on the diagram surface so it is in proper view.
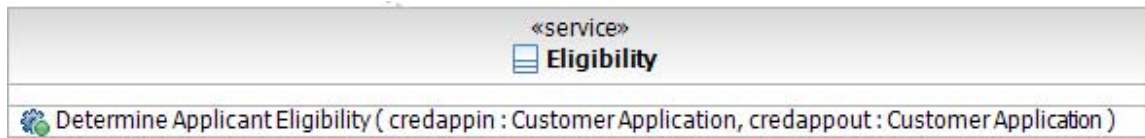
___xii.   Add the parameter to the diagram

    ___a.   Select the Account Opening -> Models -> <<BusinessUseCaseModel, BusinessAnalysisModel>> Account Opening -> <<DataCatalog>> Business items -> **Customer Application** business entity then drag and drop it on the diagram.

___xiii.   Add the "**Message**" business entity to the diagram

___xiv.   Press **Ctrl+S** to save the model

Here's how your diagram should appear:
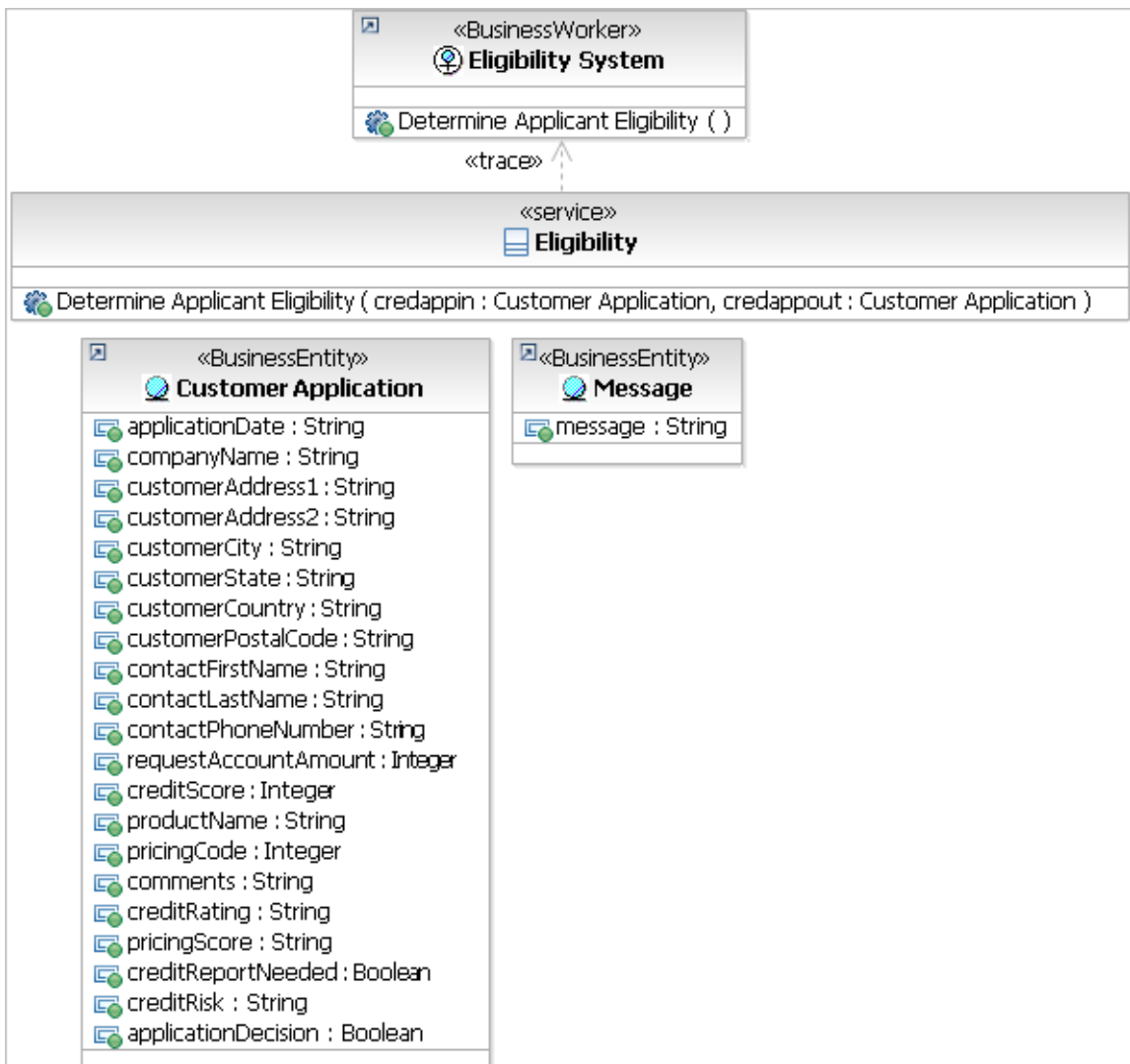
Generate Initial Implementation

Apply Transformations

At this point the Architect wants to use an RSA transformation that automatically generates the skeleton J2EE implementation from our UML design model.

The transformation recognizes several UML model elements and generates appropriate J2EE code based on their stereotype. We will load a UML profile that contains an optional <<Service>> stereotype that the transformation will convert into a Session EJB.

___xv.   Create the transformation configuration

       ___a. Select **Modeling -> Transform -> New Configuration** menu item

       ___**b.** Name the new configuration **UML to EJB for Account Opening**

       ___c. Select **UML to EJB**
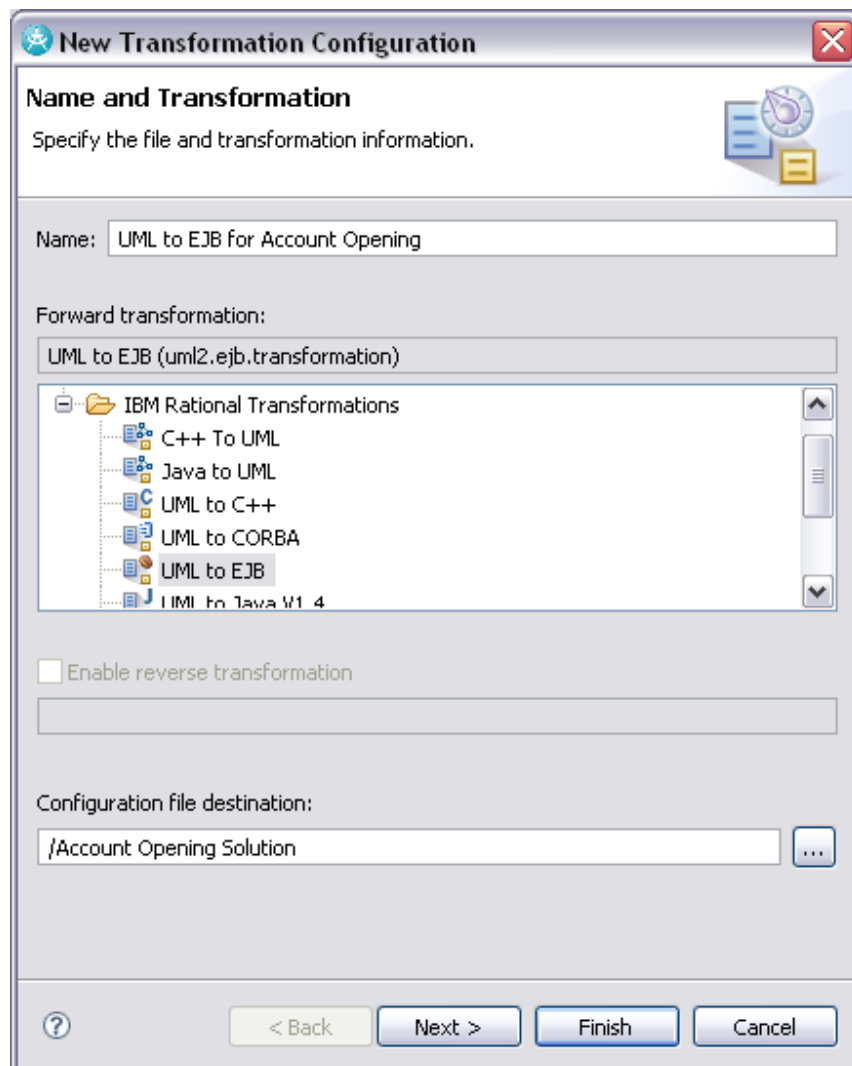
___d. Use the **…** button to select **Account Opening Solution** as the Configuration file destination.

___e. Click Next >
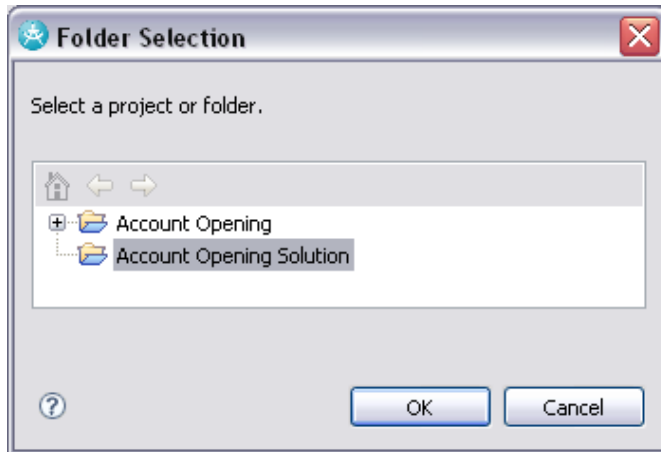
___xvi. Specify the source

___a. Select **Account Opening Solution -> Models -> Design Model** as the source
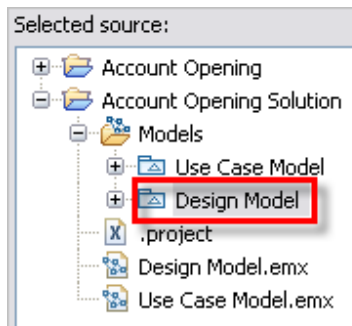
___xvii. Specify the target

___a. Click Create New Target Container…

___**b.** In the New EJB Project window, type **Eligibility** as the Project Name then click **Finish.**
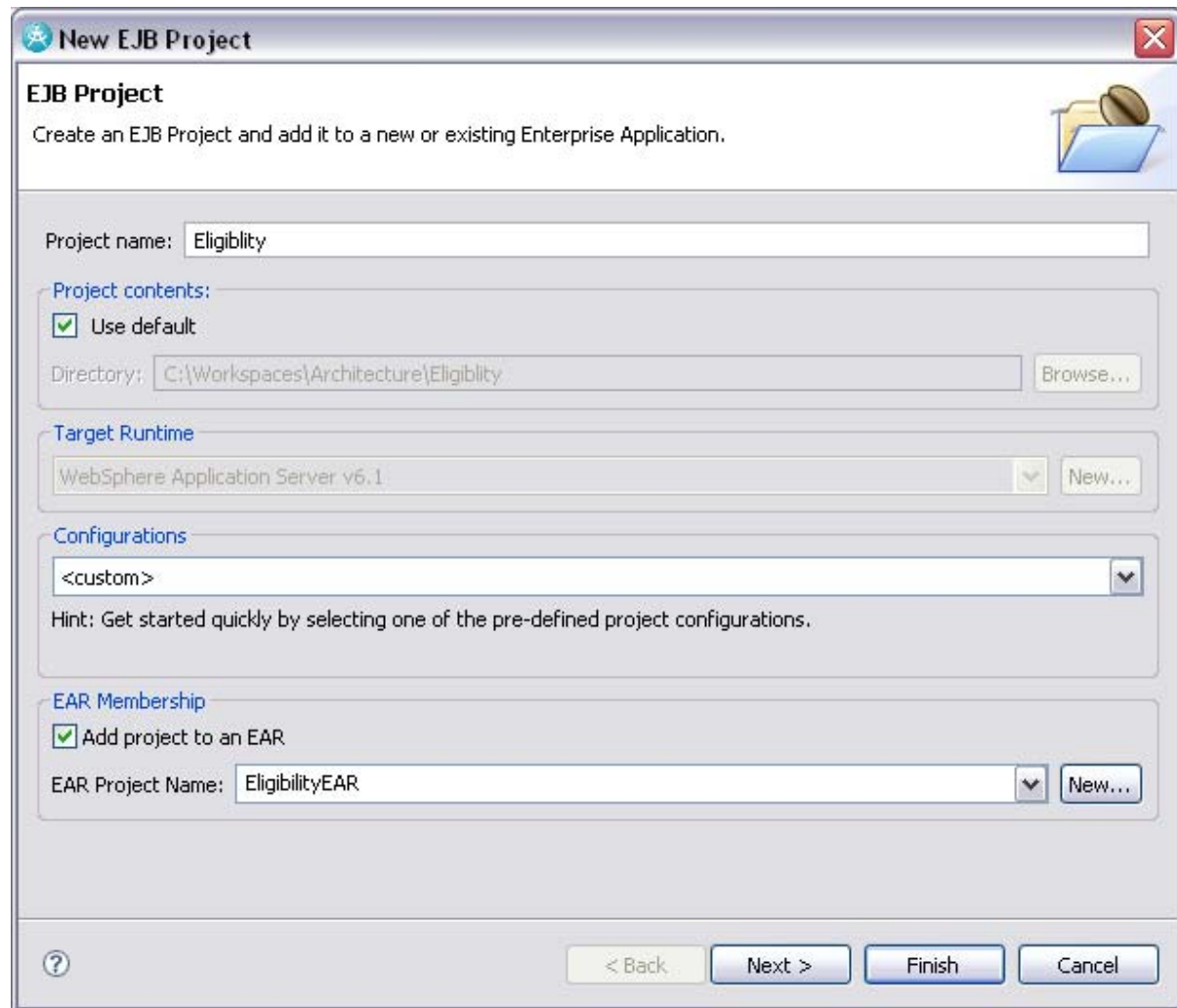
___xviii. Complete the configuration

    ___a. Click [Finish]

___xix.   Generate the skeleton code for the service and the parameter

    ___a. Select the **Eligibility** service, the **Customer Application** and the **Message** business entites in the Eligibility Service Diagram class diagram by holding down the Ctrl key and left-clicking each class once.

    ___**b.** Right-click over one of the selected classes then select **Transform -> UML to EJB for Account Opening.tc (UML to EJB)**
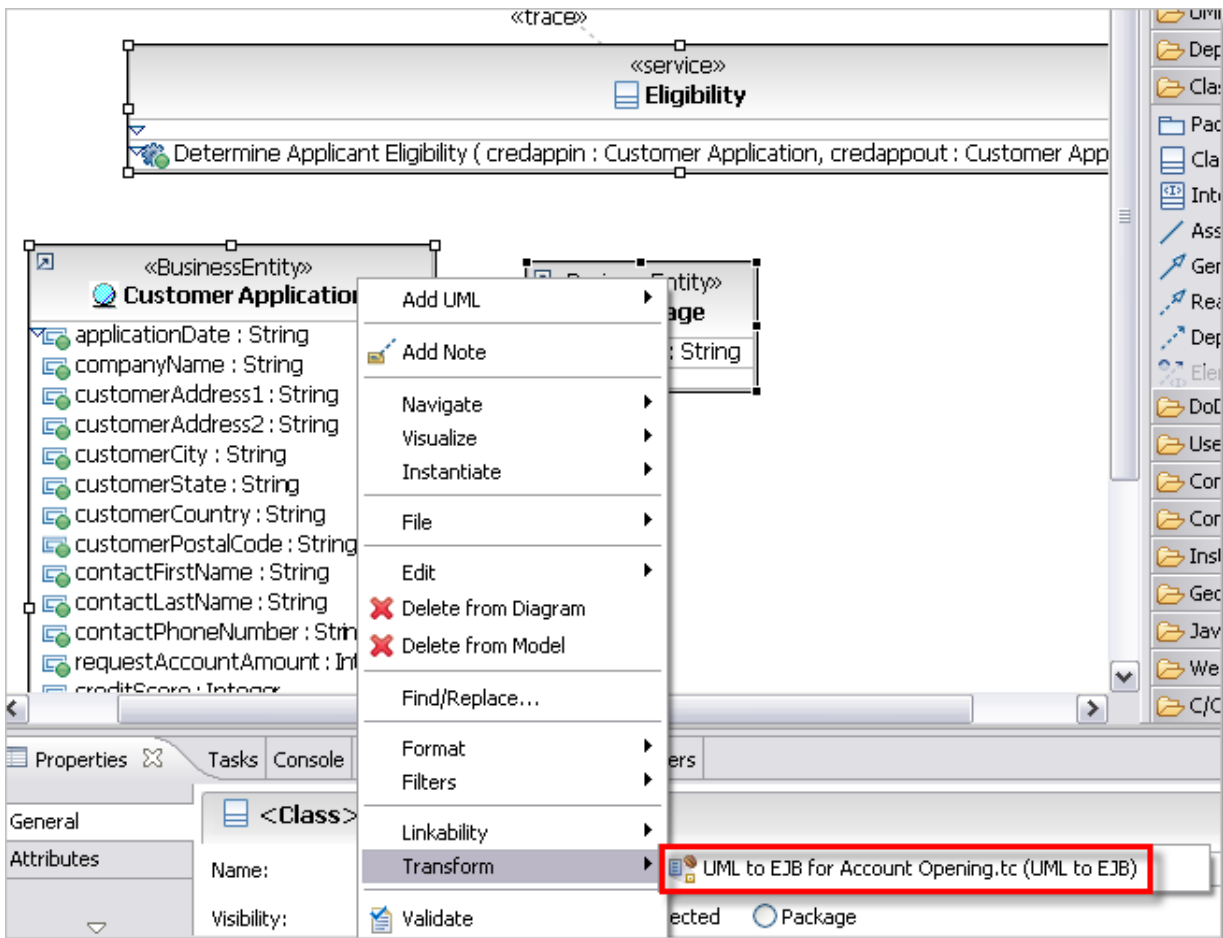
___xx.   After the transformation is complete, browse the generated code for the service
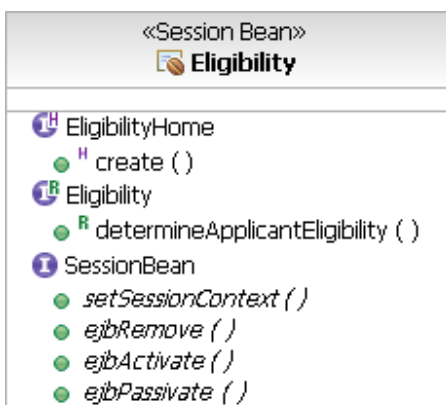
___**a.** Right-click on the **Eligibility** session bean in the new default.dnx diagram and select **Navigate -> Open with -> EJB Bean Java Editor**

___b. Right-click on the white background of the java editor and select **Show in -> Package Explorer**
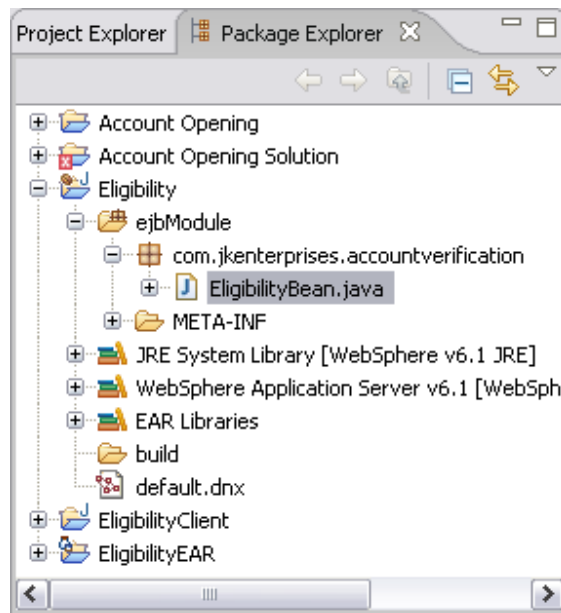
___xxi.   Browse the generated code for the parameter

___a. In the Package Explorer, browse to **Eligibility Client -> ejbModule ->
BusinessItems**
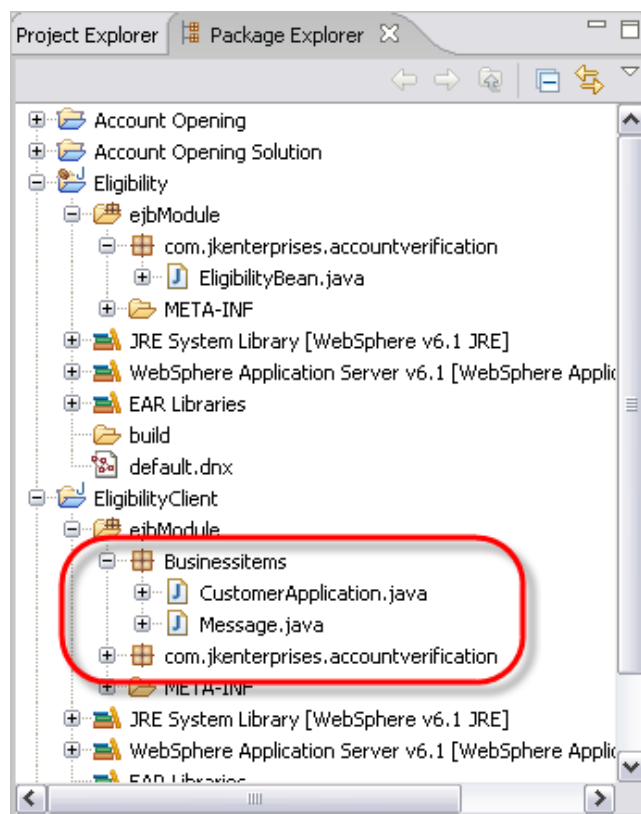
___b. Double-click on the **CustomerApplication.java** file to view the Java code generated for this entity.

___xxii. **Close** Rational Software Architect (RSA). Click **Yes** to save any changes.

Congratulations! You have successfully completed Part 1.

Service Creation

Module Introduction

In this module the student plays the role of the Developer. Starting with the skeleton implementation produced by the Architect, the Developer uses Rational Application Developer to refine the implementation and to add the required business logic. The Developer then generates a Web service from the implementation, deploys and unit tests the service, passing parameters and verifying that everything works as expected. The Developer then makes the implementation available to the Integration Developer.

Module Overview

The module is divided into four parts. These parts correspond to a set of activities that the J2EE Developer performs in a typical Model Driven Development approach.

- **3.1**:   Refine the Implementation
- **3.2**:   Deploy the Application
- **3.3**:   Generate the Web Service
- **3.4**:   Unit Test the Web Service

> **Important**: *If you encounter problems during the course of this exercise, please contact the class instructor or any of the module assistants.*

> **Note**: *This module is intended to be completed in **30 minutes**.*

Refine the Implementation

The Architect created the design model and applied a J2EE™ transformation to generate the initial J2EE implementation. Our job as a Developer is to refine that initial implementation to match the business logic for our service.

Start Rational Application Developer

The Developer starts Rational Application Developer after the Architect created the initial implementation of the Eligibility service. The Developer uses a new workspace, in which the projects created by the Architect were already imported.

In terms of functionality, Rational Application Developer is a subset of Rational Software Architect. As your laptop has RSA installed, we will launch RSA in this module as well, but we will only use features that are available in RAD.

- Start RAD

    - Double-click on the Rational Application Developer shortcut on your desktop

Add the business logic

The Developer adds business logic to the skeleton implementation generated by the architect. To keep the module simple, we created a code template, called **dae** (Determine Applicant Eligibility) that initializes the parameters and determines whether the credit report is needed or not simply based on the company name.

- The Developer adds the implementation

    - If you are not in a J2EE Perspective, open a J2EE Perspective.

> *The Tasks view appears at the bottom of the workspace. When it generates codes, RSA will put tasks into the Task list as reminders that we may have to modify the generated code. The first task (TODO Auto-generated method stub) was created when RSA generated the initial J2EE implemented. In the Task list, double-click on the "TODO Auto-generated method stub task" to open and position the editor.*

- If you don't see the Tasks view, select: **Window -> Show View -> Tasks**

- Double-click on the task **TODO Auto-generated method stub**

Problems	Tasks ✕	Properties	Servers	Database Explorer	C

1 items

✓	!	Description	Resource
		TODO Auto-generated method stub	EligibilityBean.java
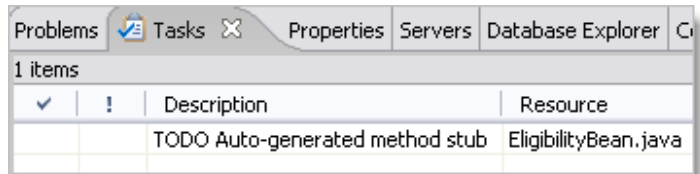
- Right-click anywhere inside the Java code editor and select **Show in ->Package Explorer**

Project Explorer	Package Explorer ✕

- ⊟ Eligibility
  - ⊟ ejbModule
    - ⊟ com.jkenterprises.accountverification
      - ⊟ EligibilityBean.java
        - ⊞ EligibilityBean
    - ⊞ META-INF
  - ⊞ JRE System Library [WebSphere v6 JRE]
  - ⊞ WebSphere Application Server v6.0 [WebS
  - Runtime Classpath Providers
  - EAR Libraries
  - default.dnx
- ⊞ EligibilityClient
- ⊞ EligibilityEAR

- The Package Explorer view shows the location of the EJB™ Bean class
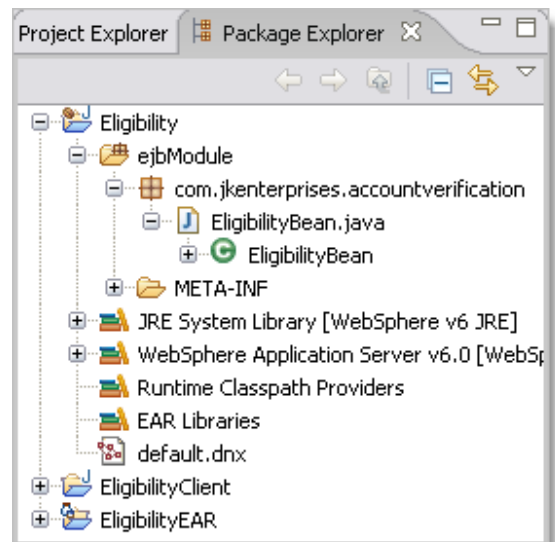
- Select these two lines (the **comment** line and the **return** line)

```
public CustomerApplication determineAppl
       CustomerApplication credappin) {
    // TODO Auto-generated method stub
    return null;
}
```
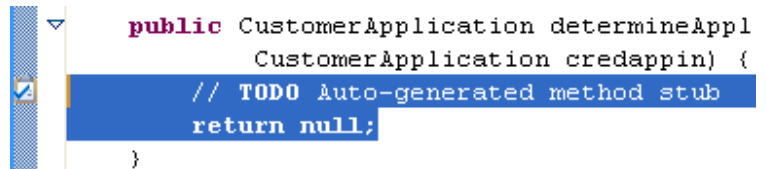
- Type **dae**, then press **Ctrl+SpaceBar** to access the code template

```
public CustomerApplication determineAppl
       CustomerApplication credappin) {
dae
}
```
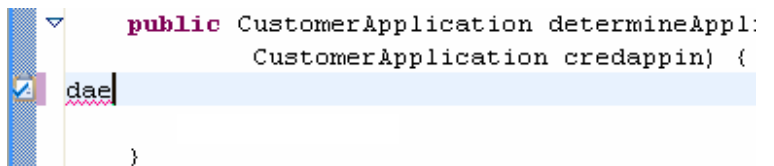
- Select the code template

```
dae
```
| dae - Determine Applicant Eligibility logic |
| DaemonConnectException - org.eclipse.hyades.execution.cor |
| DaemonCoordinator - com.ibm.ws.scheduler |
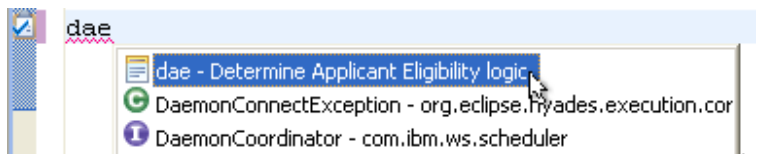
- Resolve the imports automatically

  At this point you will have to resolve the import for the `DateFormat` Java type:

  `String date = DateFormat.getDateInstance(DateFormat.MEDIUM).format(new`

    - Right-click anywhere in the Java code editor and select **Source -> Organize Imports**

    - Select **java.util.Date** and click `Finish`

    - Type **Ctrl+S** to save the EligibilityBean.java file

Deploy the Application

In this exercise we will create, deploy and run our **Eligibility** web service.

The Eligibility web service is one little piece of our bigger application – the Account Verification – that the Integration Developer is putting together.

After adding the required business logic, the J2EE Developer will convert the session bean into a web service, test it, and will make it available to the Integration Developer.

Start the Application Server

We will first start the server and then deploy the project (inverting the order of these two steps will start the server in debug mode).

___xxiii. Start the WebSphere Application Server server

  ___a. In the **Servers** view, start the WAS Server using the icon
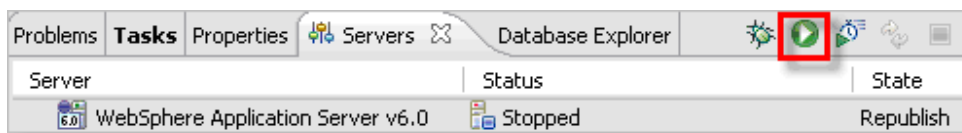
  ___b. Wait a couple of minutes until the server status is **Started** and the state is **Synchronized**.
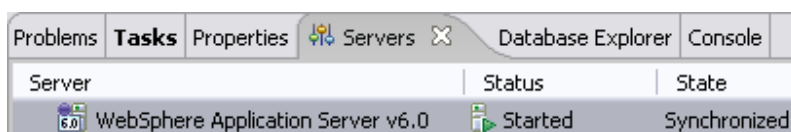
___c. Check that the server started correctly, looking at the **Console** view:

```
SchedulerServ I   SCHD0077I: The Scheduler Service is starting the Schedulers.
RMIConnectorC A   ADMC0026I: The RMI Connector is available at port 2809
SchedulerServ I   SCHD0078I: The Scheduler Service has completed starting the S
WorkSpaceMana A   WKSP0500I Workspace configuration consistency check is false.
WsServerImpl  A   WSVR0001I: Server server1 open for e-business
```

> *The Console view in RSA may display SystemOut messages from other WAS 6.0 applications which run whenever the application server is started. These messages can be ignored.*

Deploy the Project

___xxiv. Add the Eligibility project to the WAS server

___**a.** In the **Servers** view, right-click **WebSphere Application Server v6.0 -> Add and remove projects …**

___b. Select **EligibilityEAR**, click `Add >`, then `Finish`

Publishing to WebSphe...1...: (7%)
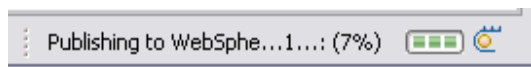
____**c.**   Wait until the workspace is built and the EAR is published
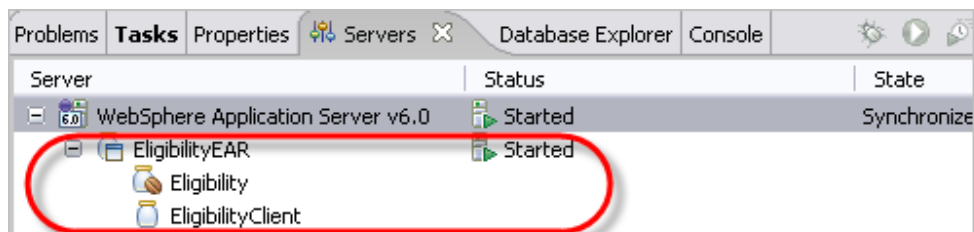
Generate the Web Service

In this part we will create and run a web service based on our **Eligibility** session bean.

Generate the Web Service

The Developer creates a web service from the Eligibility session bean

___ a.   Select the **Eligibility** session bean

___d. Switch to the **Project Explorer** view

___**e.** Select **Eligibility -> Deployment Descriptor: Eligibility -> Session Beans -> Eligibility**
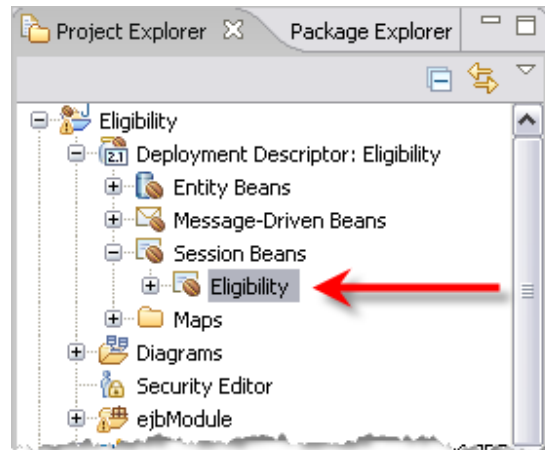
___xxv. Execute the Web service wizard

___a. Right-click and select **Web Services -> Create Web service**
[if you don't see this menu, then your Web Services role is not enabled. Go back to Exercise 1.1 step 2]

___b. On the Web Services panel, ensure **Bottom up EJB Web Service** is selected as the Web service type.
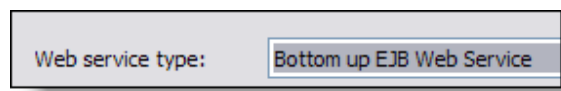
___c. Drag the slider bar up to the "**Test**" level in the "**level of service generation**" box. This is the topmost slider in the window.

___d. Select **Publish Web service** then click the **Next** button.
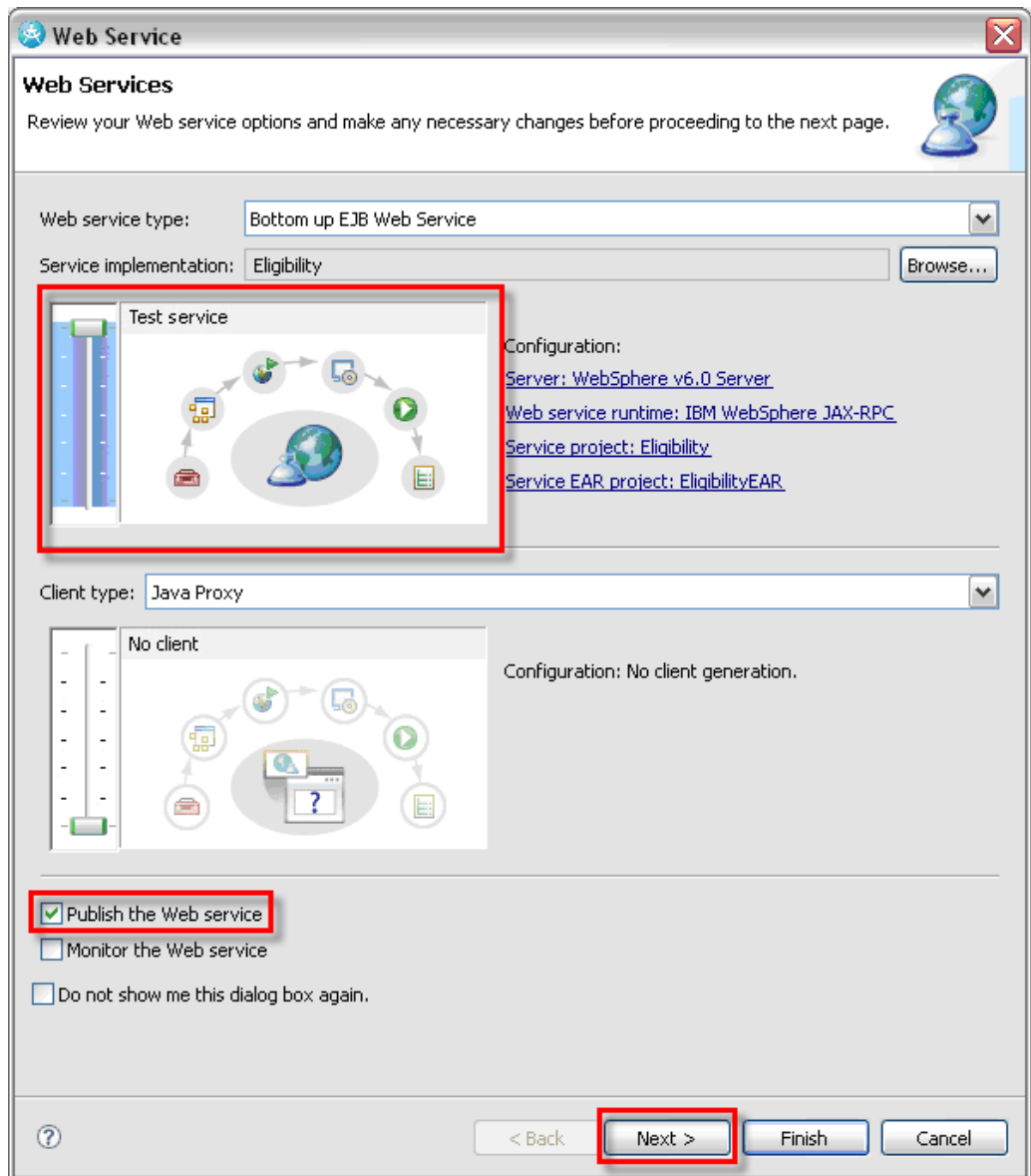
___e. On the **Web Service EJB Configuration** panel, note that RSA will create a Web application called **EligiblilityHttprouter** project for us. This is the project where the Web service will be generated.

Router Project: EligibilityHttpRouter
Http SOAP action: operation

Click the **Next** button.

___f. On the **Web Service Java Bean Identity** panel, note that if our bean had more methods, we could pick which method to expose as an operation on our service. The only method we have is automatically selected.

Check the defaults:

WSDL file:	Eligibility.wsdl
Style and Use:	Document/Literal
Security Configuration:	No Security

Click the **Next** button.

___g. **Wait** for the Web Service to be created.

___h. Click the **Finish** button in the Test Web Service window.

___i. **Wait** for the Web project to be deployed and executed.

When the **Web Services Explorer** window appears, the service is ready to be tested.

Inspect the Generated Artifacts

Before testing our service, let us look at what was generated by Rational Application developer.

In the **EligiblityHttpRouter project** folder, you can see the **Eligibility Servlet** web service.

[OPTIONAL] Unit Test the Web Service

Enter Parameters and Execute the Web Service

- Switch back to the **Web Services Explorer** view.

-

___xxvi. In the .... Navigator view, select .... the
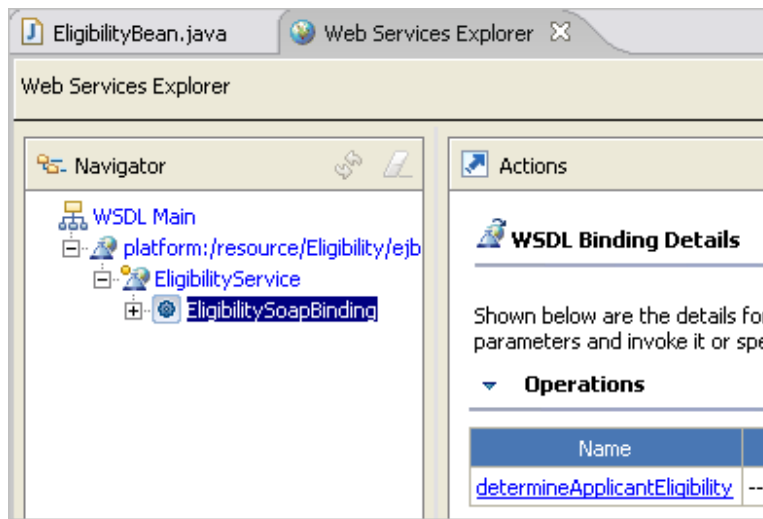
determine_Applicant_Eligibility operation

*Now we can invoke the Web service operation. The **Actions** pane should show the parameter names for the Web service.*

___xxvii. .... Add values to the parameters in the Actions window, paying attention to their types (string and integers)

___xxviii. **Add values to all the remaining parameters** (if some parameters values are missing, the Web Services Explorer will generate a validation error when you call the service)

___xxix. Select **Go** at the bottom of the Actions window. This will execute the Web Service.

Go   Reset
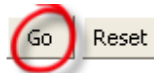
> ⚠ *Do not click the Go button a second time. Be patient. It may a minute or so for the results to display in the Console view.*

___xxx. Check that the **Console** windows shows the following message (we used the SystemErr message type, as the red text is easily distinguished):

```
SystemErr     R Determine Applicant Eligibility - begins...
SystemErr     R Credit report is not needed
SystemErr     R Determine Applicant Eligibility - ...ends
```

___xxxi. In the Actions pane, change the **companyName** parameter to **ACME** and again press the **Go** button at the bottom of the "Actions" pane.

___xxxii. The **Console** window will show that the ACME company requires a credit score.

```
SystemErr     R Determine Applicant Eligibility - begins...
SystemErr     R Credit report is needed
SystemErr     R Determine Applicant Eligibility - ...ends
```

___xxxiii. Remember to remove the running *EligibilityEAR Project* from the WAS server by right-clicking on WebSphere Application Server v6.0 in the Servers view then selecting **Add and Remove Projects…**

___xxxiv. In the *Add and Remove Projects* window, click the **<< Remove All** button then click **Finish**.

___xxxv. Finally, stop the WAS server using the **Stop the Server** icon

Problems	**Tasks**	Properties	🖧 Servers ✕	Database Explorer	Console	
Server				Status		State
⊞ 🖥 WebSphere Application Server v6.0				🖳 Started		Synchronized
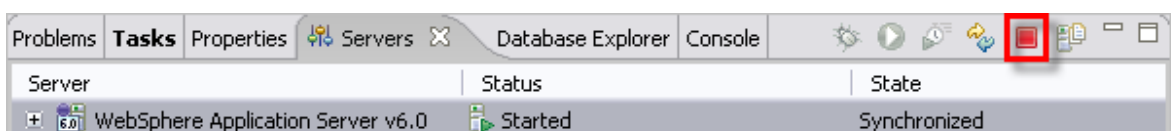
___xxxvi. Close Rational Application Developer (RAD).

Congratulations!

You have successfully completed Part 2.

Please wait for the next lecture before proceeding to the next lab.

14 Lab   Information as a Service – Federated Service

Uses VMWare Image 3
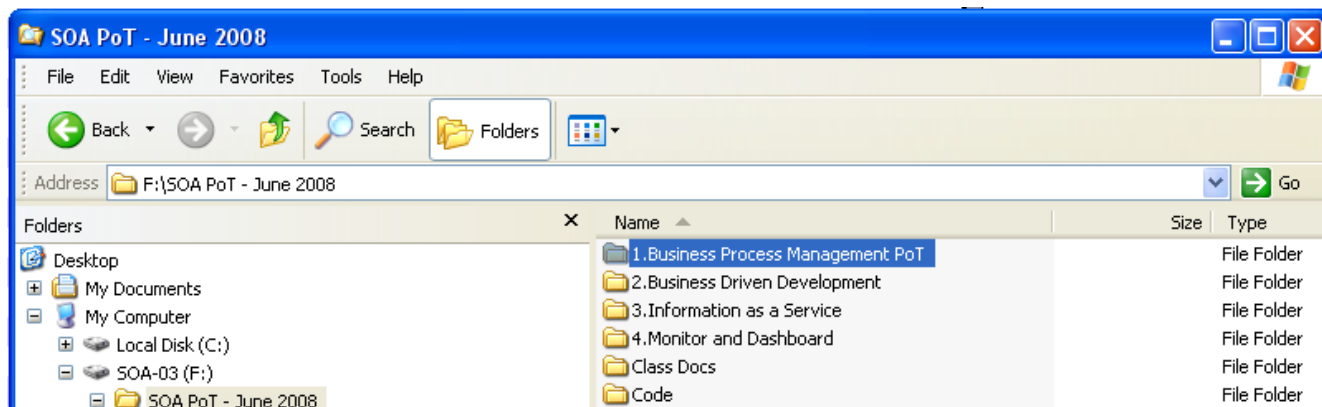
Start the "3.Information as a Service" image

Scenario Description

The scenario used for the labs is based on a fictional retail company ClipsAndTacks Office Supplies Ltd., which has been tasked to upgrade their customers' order handling process. The solution will grow from an existing batch-oriented and marginally flexible order handling process into a service-oriented, end-to-end process management based on SOA; the solution will support business value through operational flexibility and reuse.

The company wants to offer its customers with the capability to order goods online by looking at the product catalog available on the website. New customers should be able to sign up by filling out a registration form and a customer id would be automatically issued; as part of the signup process, IBM WebSphere QualityStage is used to validate the information entered by the new customer through an automatic web service call.

Once the order has been submitted, it goes through a business policy engine (to be implemented as a business rule in WebSphere Integration Developer) and if approved, customer's credit rating is checked by invoking a web service which retrieves the credit score from an XML report stored in a DB2 database.

IBM TechWorks

Software Group   IBM

SOA Demo Architecture
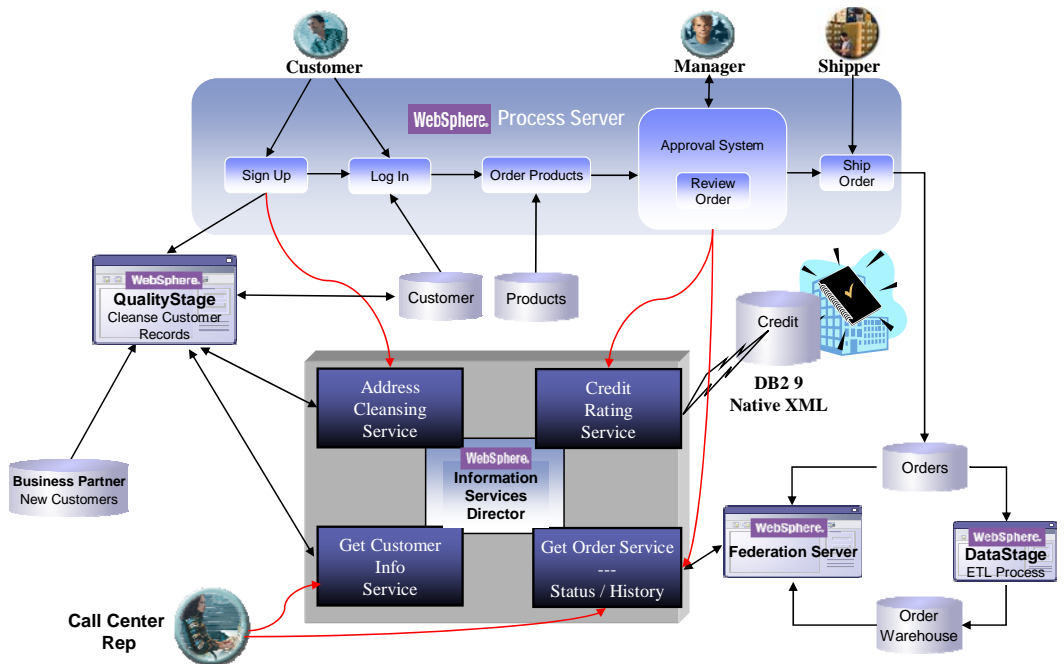
If the credit rating is acceptable, the order is sent for shipping and the order details are stored in the operational database (Informix®). This data is then periodically moved to a data warehouse (DB2) through WebSphere DataStage. In case the credit rating is not acceptable, the order is sent to a manager for review. The manager can then look at the order details and pull out the customer's order history from the operational database as well as the data warehouse. This is done through a call to an Order Service which is implemented using WebSphere Federation Server as the Information Provider. Based on the information received, the manager may approve or decline an order.

The design of this order handling process that we just described typically starts with a business analyst, who models the process using WebSphere Business Modeler. The application developer then uses WebSphere Integration Developer and WebSphere Process Server to convert the business model into a running process. The various services are then invoked as part of that business process. The new version of WebSphere Integration Developer enables the application developer to select WebSphere Information Server as one of the services enabler.

Order Handling Process – WebSphere Business Modeler

Note: The Business Modeling phase is outside the scope of this Proof of Technology.
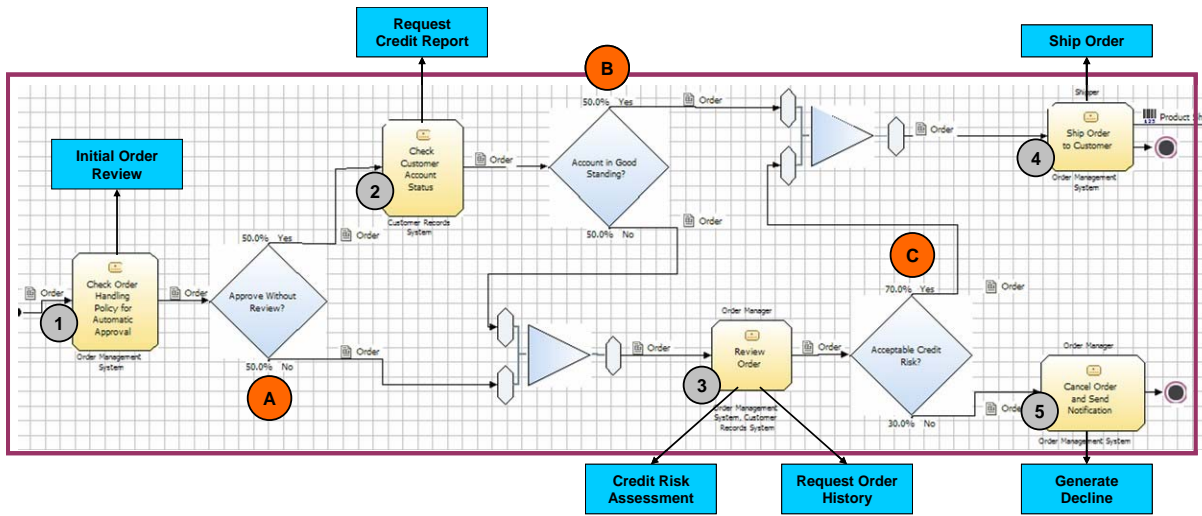
Key Decision Points

A. **Is the order amount below the specified limit to approve the order directly?**

B. **Is the credit rating acceptable?**

C. **What is the credit risk?**

Implementation type

- **Order Handling Process - Rule Group**
- **Check Customer Account Status - Java Task**
- **Review Order – Human Task (Manager)**
- **Ship Order – Human Task (Shipper)**
- **Cancel Notification – Human Task**

IBM *TechWorks*

Lab Setup

The labs are based on the content of a VMWare image that includes the following software:

- IBM WebSphere Business Modeler

- IBM WebSphere Integration Developer (version 6.0.2)

- IBM Information Server (version 8), which includes:

  - IBM WebSphere DataStage and QualityStage

  - IBM WebSphere Federation Server (including IBM DB2 9)

  - IBM WebSphere Metadata Server

  - IBM WebSphere Information Services Director

  - IBM WebSphere Application Server

- IBM WebSphere Process Server (version 6)

- Informix Dynamic Server (version 10.00 TC5)

Userid and password

For ease of use, we will use the same userid and password throughout the exercises.

- Userid: **administrator**

- Password: **ibmpassw0rd**   (Note: zero, not the letter o)

This is the only userid and password you need both to start the VMWare image, and to use any software component that prompts you for this information.

Start required Services

To complete the labs successfully, you should have a number of services up and running. You can run the batch files named ProcessServer.bat and OtherServices.bat stored in C:\SOAdemo\Scripts\Scenario1, or double-click on the 2 icons placed on the desktop. The services and applications that are being started by running these 2 jobs are:

- IBM WebSphere Process Server

- ASB Agent

- DataStage Engine Resource Service

- DataStage Telnet Service (this service is not needed as all components run on the same machine)

- DB2 (DB2 – DB2COPY1 – DB2 , and DB2DAS –DB2DAS00)

- Informix Dynamic Server Message Service

- Informix IDS – ol_pistew

- IBM WebSphere Application Server V6 – pistewNode02

Note that in our lab configuration, all the components (DataStage, Console for IBM Information Server, IBM WebSphere Application Server and ASB Agent) are located on one single machine. This configuration would not be typical in a customer environment as it would adversely affect performance.

Short overview of the lab exercises

- Lab setup and initialization: in this lab,

  - you start the VMWare image,

  - you start the Windows® services required by the software products used in all the labs,

  - you use the Console for IBM Information Server – which provides the functionality of IBM WebSphere Information Services Director -- to configure Information service Providers (such as IBM WebSphere DataStage and QualityStage, or IBM WebSphere Federation Server), and to create an Information Services Application. This initial configuration setup is needed before you can start creating and deploying information services.

- Scenario 1: The objective of this scenario is to learn the steps required to expose a pre-existing DataStage job as a service.

  - Use IBM WebSphere DataStage and QualityStage Designer to modify an existing DataStage batch job and enable the job for information services; the key modifications required consist of replacing the DataStage input and output stages with WISD input and output stages, and adding an XML output stage which collapses the result set into one XML string.

  - Use the Console for IBM Information Server to create a DataStage job Information Service: in doing this, you will need to name the service, to specify the bindings used by the service, to define the operation (i.e. the DataStage job) that the service executes, and verify the input and output parameters. Runtime parameters for the service can also be tuned here.

  - Use the Console for IBM Information Server to deploy the application. Remember that you always create a service in the context of an application.

  - Use the Web Console for IBM Information Server to check the WSDL file associated to the service.

  - Test the service using WebSphere Integration Developer.

- <u>Scenario 2</u>: The objective of this scenario is to learn the steps required to expose a DB2 federated query as a service

  - Use DB2 Control Center to setup a federated environment; in our scenario, we will federate data about customer orders; the most recent orders are stored in the operational database (Informix), and the data warehouse (DB2) contains consolidated orders.

  - Define the DB2 query that retrieves the orders from both the data warehouse and the operational data store.

  - Use the Console for IBM Information Server to create an Information Service that uses the federated query.

  - Use the Console for IBM Information Server to deploy the application.

  - Use the Web Console for IBM Information Server to check the WSDL file associated to the service.

  - Test the service using WebSphere Integration Developer.

- <u>Scenario 3</u>: In this scenario, we provide a service which performs data cleansing of customer information (name and address); this scenario is very similar to the first one. The job that we start from contains QualityStage stages to perform data standardization and data matching. The individual steps required to expose the job as a service are the same as the ones we have gone through in Scenario 1.

- <u>Scenario 4</u>: Before a purchase order is accepted, it may be necessary to check the customer's credit rating; this would certainly be the case if the customer just placed a substantial order. A customer credit report is typically delivered as an XML file; in our company, we have decided to store the credit reports in an XML-typed column of a DB2 table.

  - Build a query which retrieves just the average credit rating for the customer.

  - Use the Console for IBM Information Server to create an Information Service that uses the DB2 query.

  - Use the Console for IBM Information Server to deploy the application.

  - Use the Web Console for IBM Information Server to check the WSDL file associated to the service.

  - Test the service using WebSphere Integration Developer.

- <u>Scenario 5</u>: In this exercise, you will learn how to build a simple application that surfaces one of the web services. This simple Web application uses the content of an HTML form (input data) in its JSP logic to call the "client proxy" for the service. The client proxy is the code that a web application uses to call the Web service. You will use the Web Services wizard available in WebSphere Integration Developer to create a client proxy for the Customer Information Web Service.

- Scenario 6: This scenario shows how to develop the application using WebSphere Integration Developer; you will also learn how to configure the web service in WID.

Getting started with WebSphere Information Services Director (WISD)

WebSphere Information Services Director provides all the required functionality to create and deploy services. However, a few initialization steps need to be performed beforehand:

- Create Information Service Providers

- Create a Project and an Information Services Application

To perform these steps, and to create and deploy the services later on, you use the Console for IBM Information Server, which provides the functionality of IBM Information Services Director.

___9.   Start the Console for IBM Information Server.

Before starting the Console for IBM Information Server, you must have an application server up and running. In our configuration, we are using WebSphere Application Server (which was installed as part of the IBM Information Server install). Verify that the IBM WebSphere Application Server service is already started (the service name is IBM WebSphere Application Server V6 – pistewNode02); if it is not started, start the service.

___ b.   Start the Console for IBM Information Server (**Start → All Programs → IBM Information Server → IBM Information Server Console**).

___ c.   You will be prompted for a user name and a password:

- user name: **administrator**
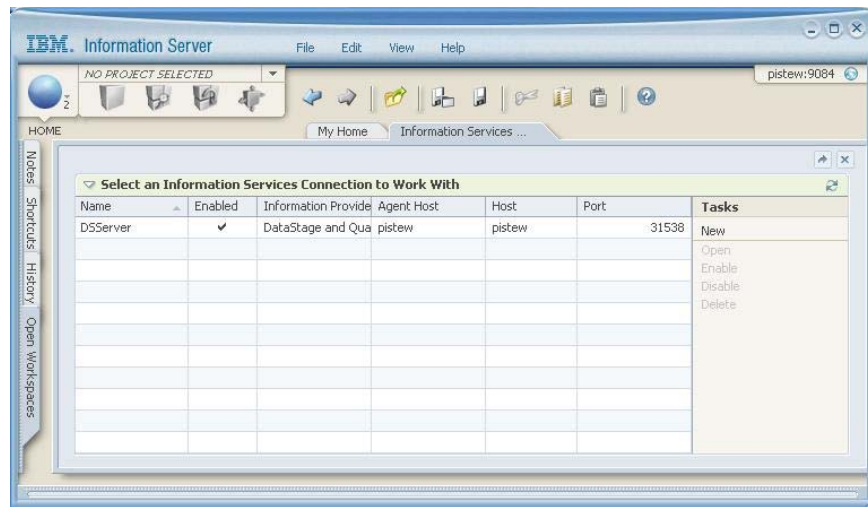
- password: **ibmpassw0rd**
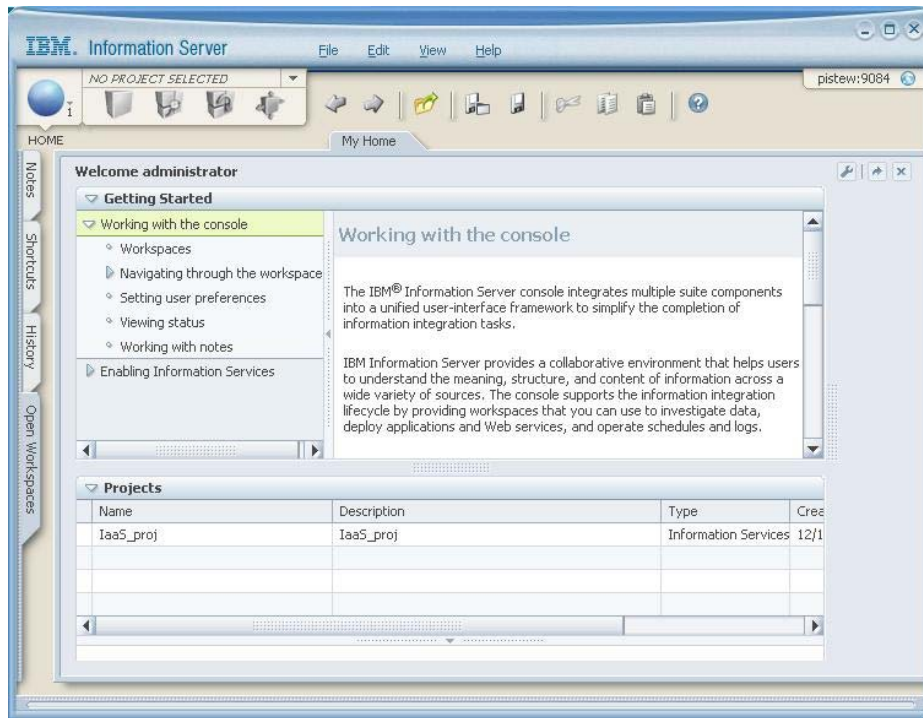
___10. Configure Information Service Providers.

An *information provider* is both the server that contains units that you can expose as services and the units themselves, such as WebSphere DataStage and WebSphere QualityStage jobs or federated SQL queries. Each information provider must be enabled. We will enable DataStage/QualityStage Server and Federation Server.

___ e. Select the **Home navigator** icon, then from the menu expand the **Configuration** folder, and select **Information Services Connections**.
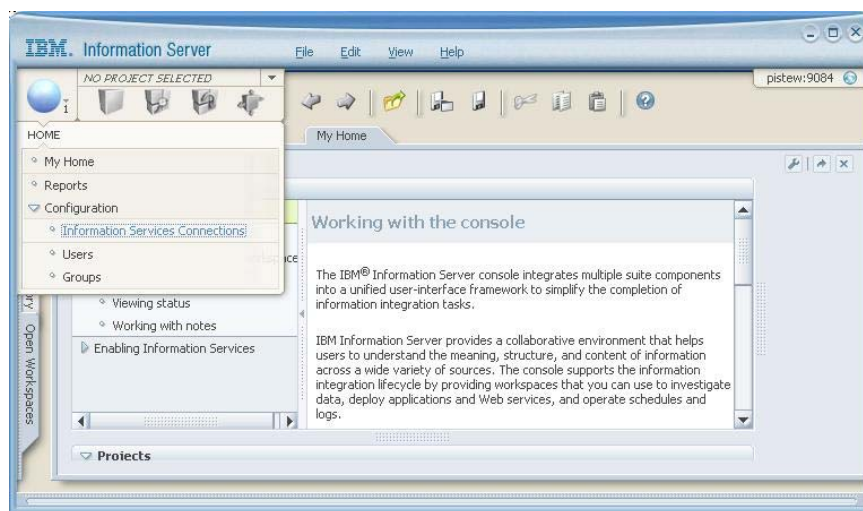

___ f. A window is now being opened with all the configured connections. On this VMWare image, connections have already been configured; as a learning exercise, we will anyway define connections, similar to the ones displayed.

___ g.   Click on **New**; specify all the required information for the new connection.

- Connection Name: **PoT_DSQS**

- Information Provider Type: select **DataStage and QualityStage**

- Agent Host: select **pistew**

- User Name: **administrator**
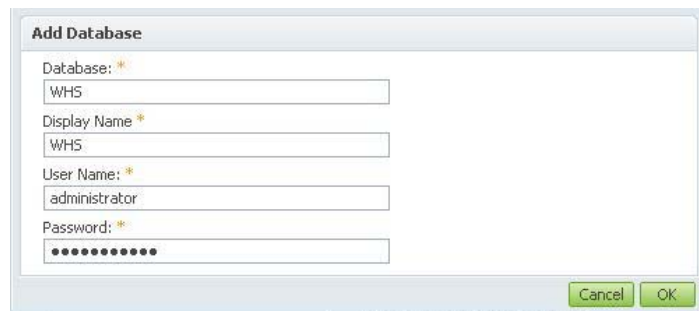
- Password: **ibmpassw0rd**
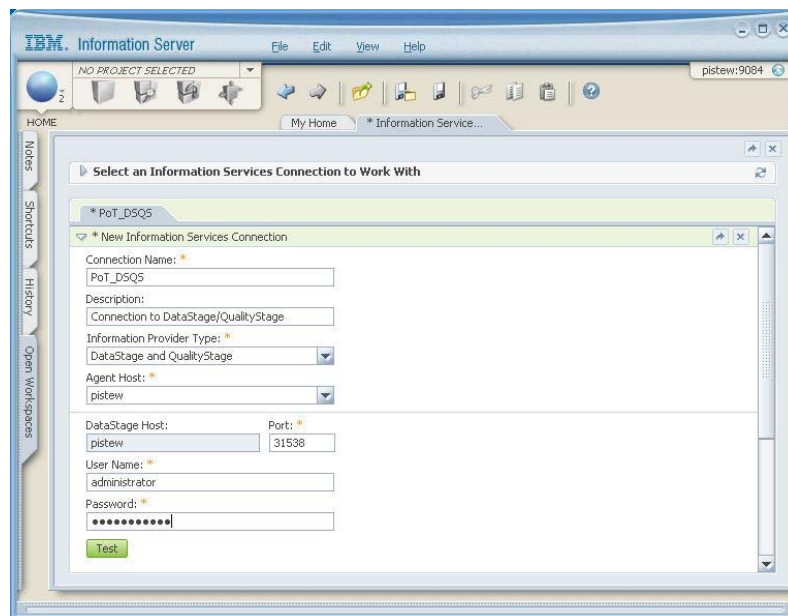
___ h.   Click on **Test** to verify that the connection is up and running. You should receive a message confirming that the validation is successful. Close the message window.

_____ i.   At the bottom of the Information Services Connection window, click on the **Save and Enable** button, then click on the Close button to close the connection window.

Now, go through the same steps to create a connection to federation server.

_____ j.   From the Console for IBM Information Server, select **Home → Configuration → Information Services Connections**.

_____ k.   In the Information Services Connection window, click on **New**; specify all the required information for the new connection.

- Connection Name: **PoT_FS**

- Information Provider Type: select **DB2 or Federation Server**

- Agent Host: select **pistew**

- Database Host**: localhost**
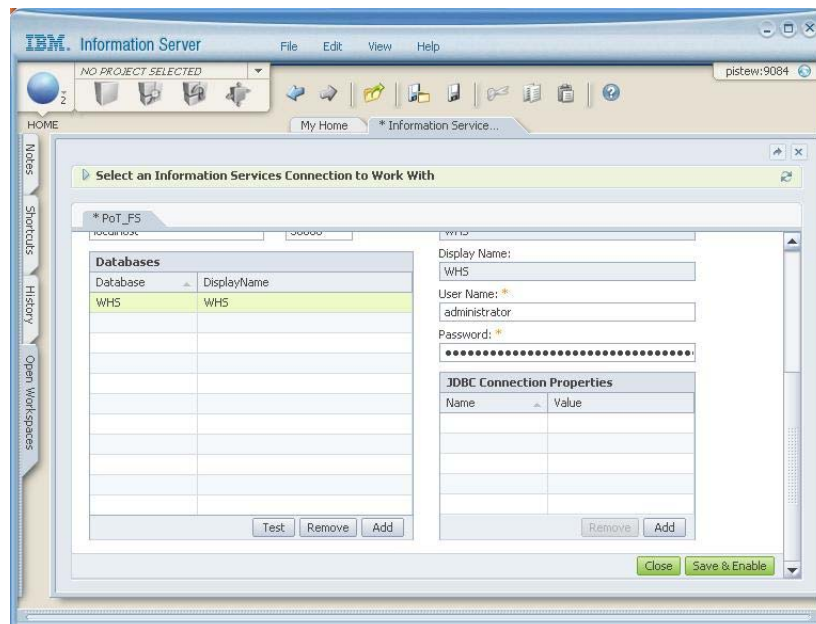
- Port: **50000**
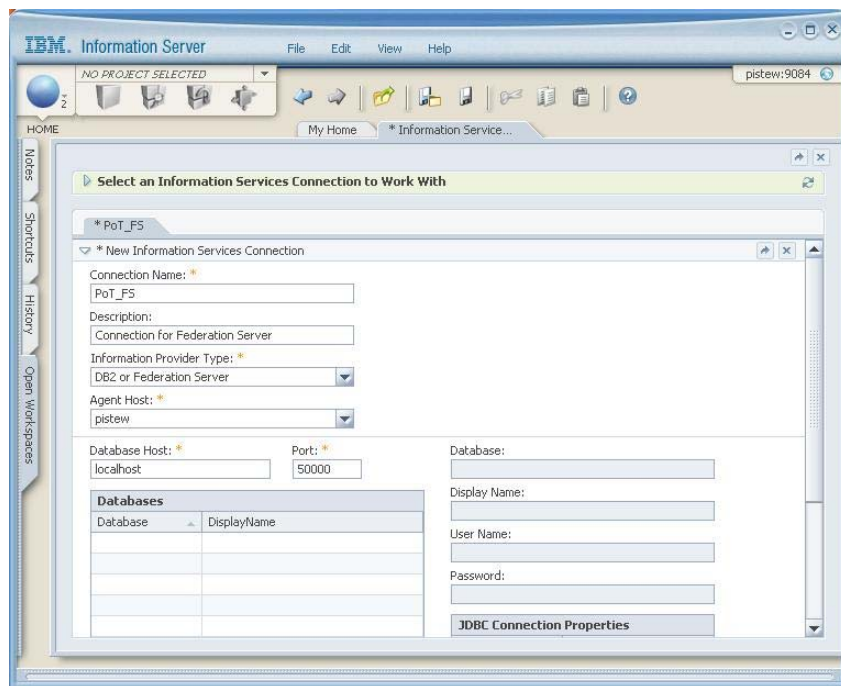
_____ l.   Click on the **Add** button, located at the bottom of the Databases section of the window.

_____ m.   In the Add Database window, fill in the fields:

- Database: **WHS**

- User Name: **administrator**

- Password: **ibmpassw0rd**


___i.   Click **OK**.

___ n.   A new row appears in the Databases table. Click on **Test** to verify that the connection to the database is successful.


___ o.   Press the **Save and Enable** button; then **close** the window.

___ p.   We have now completed the task of creating and enabling Information Services Connections. The Console for IBM Information Server displays now all the Information Services Connections that have been created.
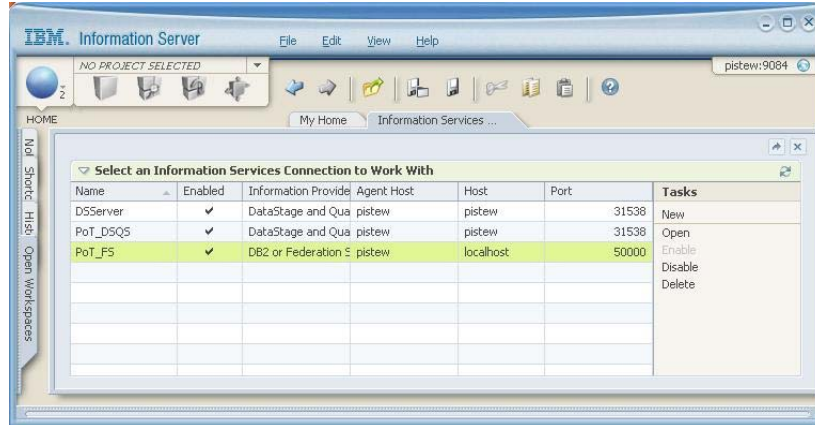

___ q.   **Close** the Information Services Connection window.

___11.   Create a Project and an Information Services Application.

___ a.   Before services can be created and deployed, we need to create at least one project and one Information Services application. One project can hold many applications, and in each application, many services can be created. For the purpose of the exercises, we will just create one project, and one application.

___ r.   Create a new project.

___xxxvii.   From the Console for IBM Information Server, select **File →
New Project**.

___ii.   Name the project **PoT_Project**.

___iii.   Here is the Project Properties window we can define users and groups allowed to work with the project, and assign project roles.
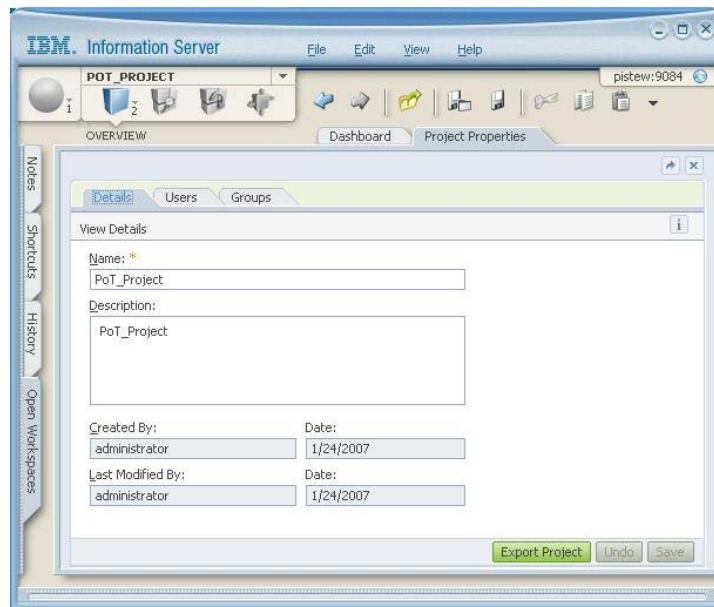
___iv.   Open the **Users** tab, select the **user administrator**, and assign to this user the two available project roles.
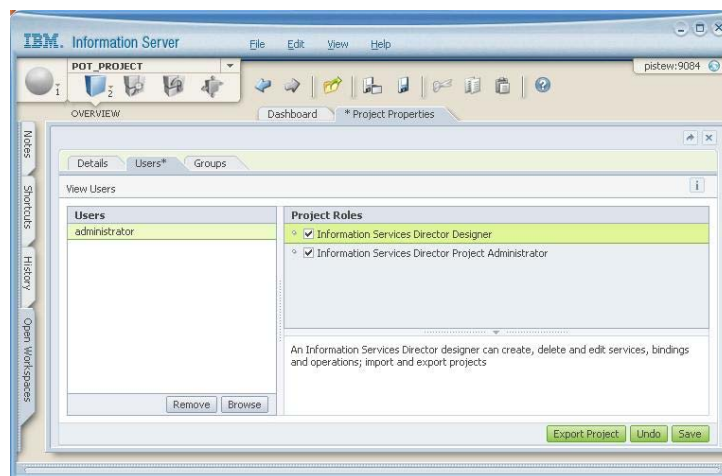

The role of Information Services Director Designer enables the user to create, delete and edit services, bindings and operations; it also lets the user import and export projects.

The role of Information Services Director Project Administrator enables the user to create and delete applications, to add and remove users and groups to projects; it also lets the user assign project level roles and set permissions on applications.

___v.   **Save** the Project definitions.

      ___vi.   **Close** the Project Properties. Click on **x** in the Project Properties window.

___ s.   Create a new application.

      ___ii.   Applications in WISD mark the deployment unit. You can include any number of services within an application.

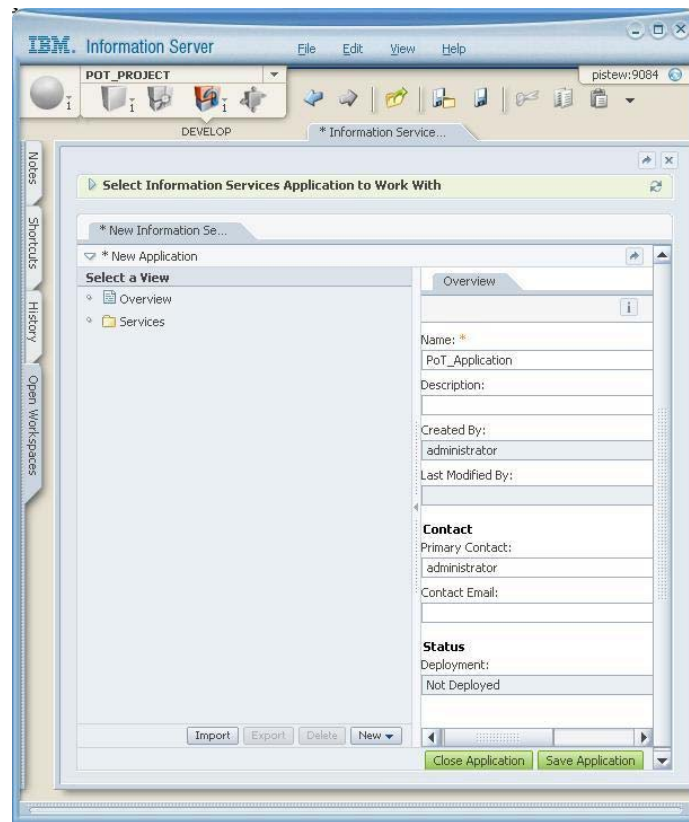   ___vii.   Open the Information Services Application window.
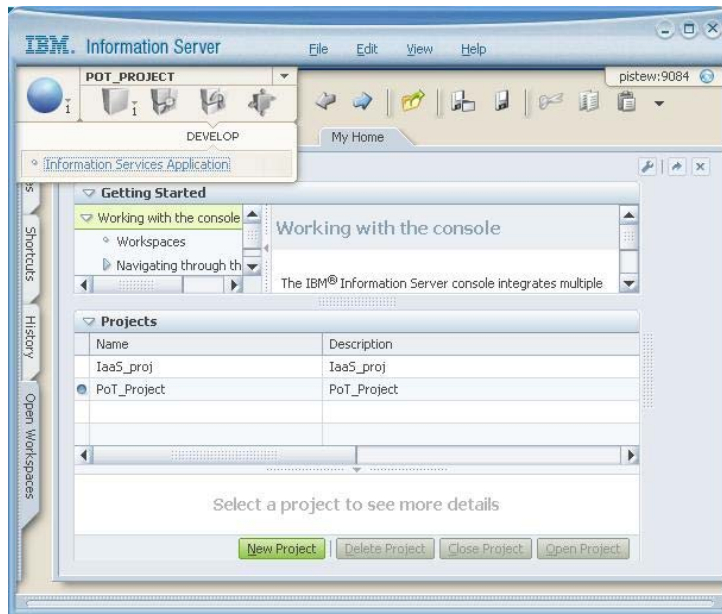
      ___viii.   Click on the **Develop** icon, and select **Information Services Application** from the menu.

      ___ix.   This opens the Information Services Applications window; it is empty right now, since no application has been created yet.
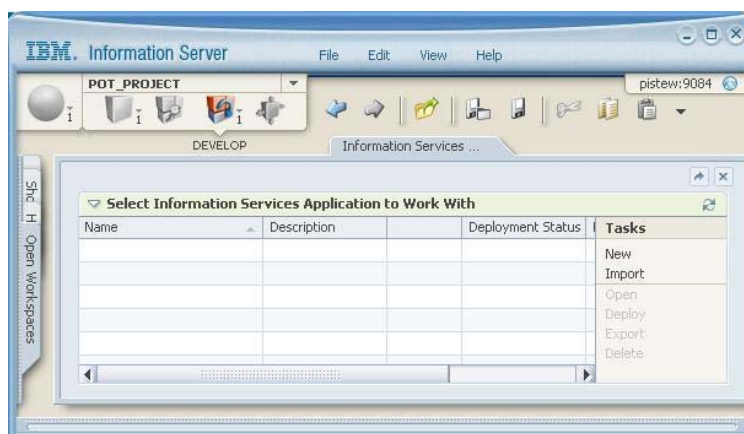
      ___x.   Click on **New** in the Task list to create a new application.

___xi. Enter **PoT_Application** in the Name field of the Overview window.

___xii. **Save** the Application.

___xiii. **Close** the Application. Now that the application is created, we can start to create services.

We have completed all the preparatory work in the IBM WebSphere Information Services Director. In the subsequent exercises, we will create and deploy services in the context of the project and application we just created.

Overview Lab Federated Service

Data Federation is the key technology delivered by the IBM WebSphere Federation Server product (previously known as WebSphere Information Integrator); it provides transparent access to diverse business information from a variety of sources and platforms as though all the data were coming from a single resource.

In our case study, the operational database ORDER which stores customer's orders resides in an Informix database while the Enterprise warehouse database (WHS) is in DB2 UDB. By completing this scenario, you will learn to set up federated access from DB2 UDB (used as the

IBM *TechWorks*

federation server) to Informix, thus accessing the information from both data sources as if it resides in a single database.

Then, you will write a federated query, and expose this federated query as a service. In the first scenario, we have used the IBM Information Server to connect to a DataStage server as an Information Provider, and to expose a DataStage job as a service. In this scenario, we will use the IBM Information Server to connect to a DB2 federated server, and to expose a federated query as a service. Note that there is another way to also expose an SQL query as a service, by using the WORF technology embedded in DB2 to do it; this method does not make use of IBM Information Server. We will not cover this alternative method here.

Part 1 – Federated Environment Setup:

Setup overview:

To enable a DB2 UDB database to have federated access to remote tables (Informix in this case) the following steps need to be performed:

1. Create Wrapper

2. Create Server

3. Create User Mapping

4. Test connectivity

5. Create nicknames

6. Test

Here is an explanation of the key concepts involved:

Wrapper:

The first step to access the remote tables through data federation is to create a wrapper to access a specific type of data source.

A wrapper is a library that allows access to particular type of data source such as Oracle, Informix, MS SQL Server, Sybase, Generic ODBC, Web Services, MQ etc. It contains information about the remote data source characteristics, and it understands its capabilities.

WebSphere Federation Server ships wrappers for many relational as well as non relational data sources. For a complete list, refer to the documentation. In addition to the IBM provided wrappers, users can create wrappers for any data source using wrapper interface specifications available from IBM.   Note that the Informix wrapper is not installed by default; in addition the code for the Informix wrapper is not done as part of the WebSphere Federation Server, but as part of the DB2 9 install. So, to get the Informix wrapper, do a DB2 9 custom install and select Informix data source support, under Server support. This feature enables the submission of distributed requests for data managed by Informix.
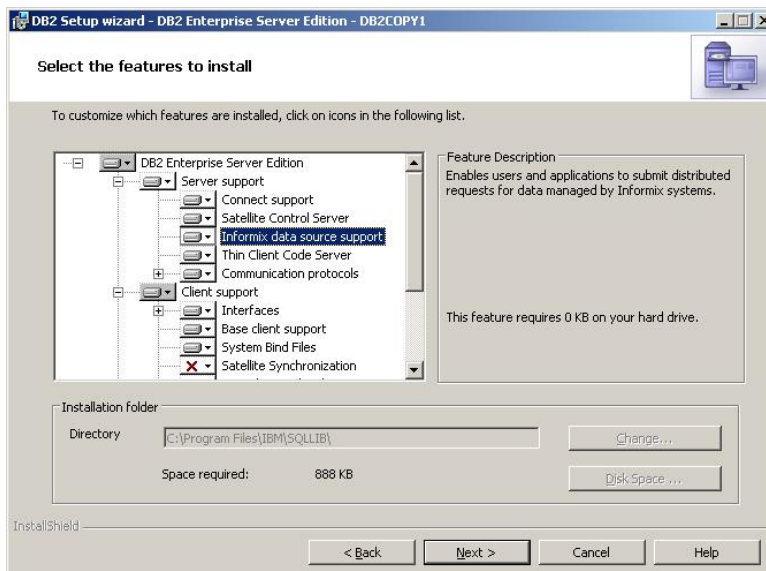
Server:

A server represents a specific data source that is accessed through a wrapper.

So for example, if you have an Oracle server running in department A and another running in department B, you will have 2 different Oracle servers, both using the same Oracle wrapper.

There are DDL statements to define a server for every data source that you want to use.

Each server can have its own set of server options. For example, server A can be defined with UserId to be in upper case, whereas server B could use mixed case UserId. Collating sequence of the remote database is another common option that is specified. This will affect the sort order used and will also help determine if parts of a SQL statement should be pushed down to be handled by the remote data source.

User mapping:

It provides a mapping from a user ID that accesses the federated server to a data source user ID and password that WebSphere Federation Server will include in connections to the remote data source on behalf of the federated user.

Nickname:

The key to WebSphere Federation Server technology is the use of nicknames.

A nickname is a local alias to data object on a remote server (mapped to rows and columns). You define a nickname for any remote data object that you want to use on a given server. In queries against your federated server, you can then use the nicknames as if they were local tables.

For example, if you have ORATAB1 as a nickname to a remote Oracle table, SQLTAB2 as a nickname to a remote SQL Server table, and DB2TAB3 as a local DB2 UDB table, a SQL statement such as the one below, will populate the local DB2 UDB table with data joined from Oracle and SQL Server tables.

INSERT INTO DB2TAB3 SELECT <o.oraclecol1>, <s.sqlservercol2>
FROM ORATAB1 O, SQLTAB2 S
WHERE <o.key>=<s.key>

In fact, the end user or application does not even need to know that they are nicknames, and that they represent remote tables or non relational data.

More significantly, the SQL operations against nicknames are OPTIMIZED by the DB2 UDB optimizer globally. DB2 UDB will determine if it more efficient to push an operation down to the remote data source, or fetch the data locally and process it within DB2 UDB. As a result, federated access using nicknames typically has very high performance.
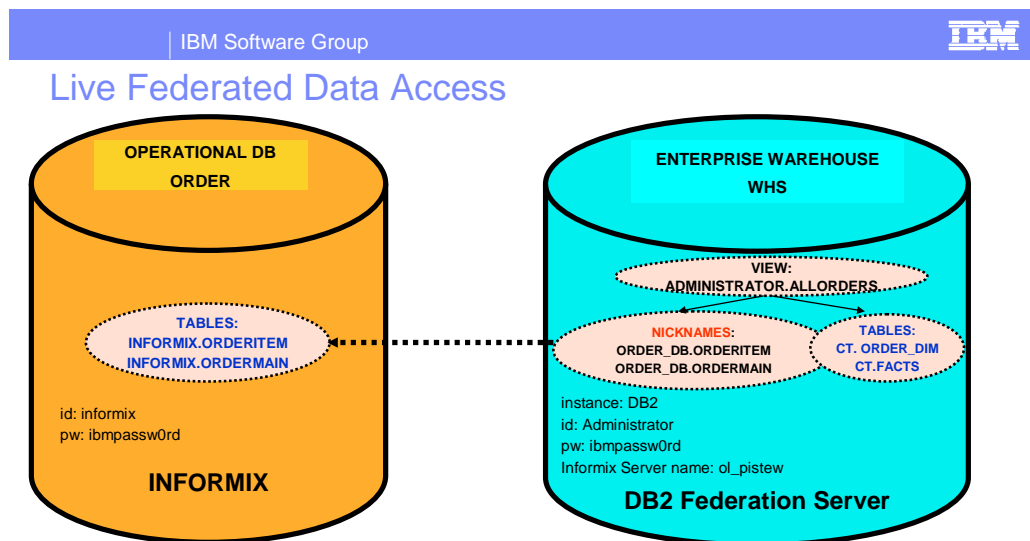
With the above as background, following are diagrams of our scenario.

DATA LAYOUT BEFORE DATA FEDERATION
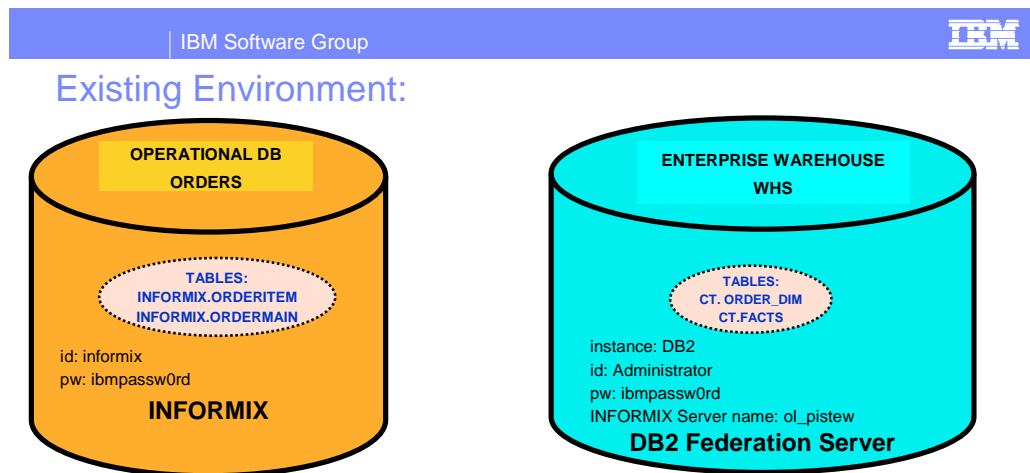
We will enable the existing enterprise warehouse database (WHS) in DB2 UDB to include federated access to tables in the ORDER database. After setting up the federated access, we should have the following layout:

DATA LAYOUT AFTER DATA FEDERATION

We will now go through the steps necessary to setup federation between WHS and ORDER databases.

The setup can either be done using SQL directly, or by using the DB2 UDB Control Center. For this lab, we will use the Control Center; however in practice it is often easier to simply modify existing scripts to setup federated access. The DB2 UDB Control Center has the option of storing the generated SQL prior to running the command; it is a good practice to save the generated SQL's so you'll be able to recreate the environment easily without using the Control Center. A set of scripts for the labs are under the directory *C:\SOAdemo\SCRIPTS\scenario2*.

IBM *TechWorks*

DB2 UDB should be set up to come up automatically, if not, start the DB2 UDB services.

Bring up the DB2 UDB Control Center,

1. Click **Start → All Programs → IBM DB2 → DB2COPY1 → General Administration Tools → Control Center**.

2. You can also bring up the DB2 UDB Control Center by typing **db2cc** from any command prompt.

Hint 1: When Control Center is launched, a preference for the display view needs to be specified. Since the federated database objects required to access data sources on the remote system are not displayed in the 'Basic' view, select the recommended 'Advanced' view.

Hint 2: Refresh the Control Center to ensure that the newly created objects are displayed. If for any reason, the new objects are not displayed even after a refresh, close the Control Center and bring it up again.)

In this scenario, we will use the **Nicknames configuration wizard** to create all the federated objects necessary to setup access to Informix tables. Bring up the configuration wizard as follows:

__1.   Open the DB2 Control Center and expand the left panel tree to display the objects under the database **WHS**.

__2.   Right click on the **Nicknames** folder and select option **Create…**

__3.   This action will bring up the Nickname Configuration Wizard, as shown below.
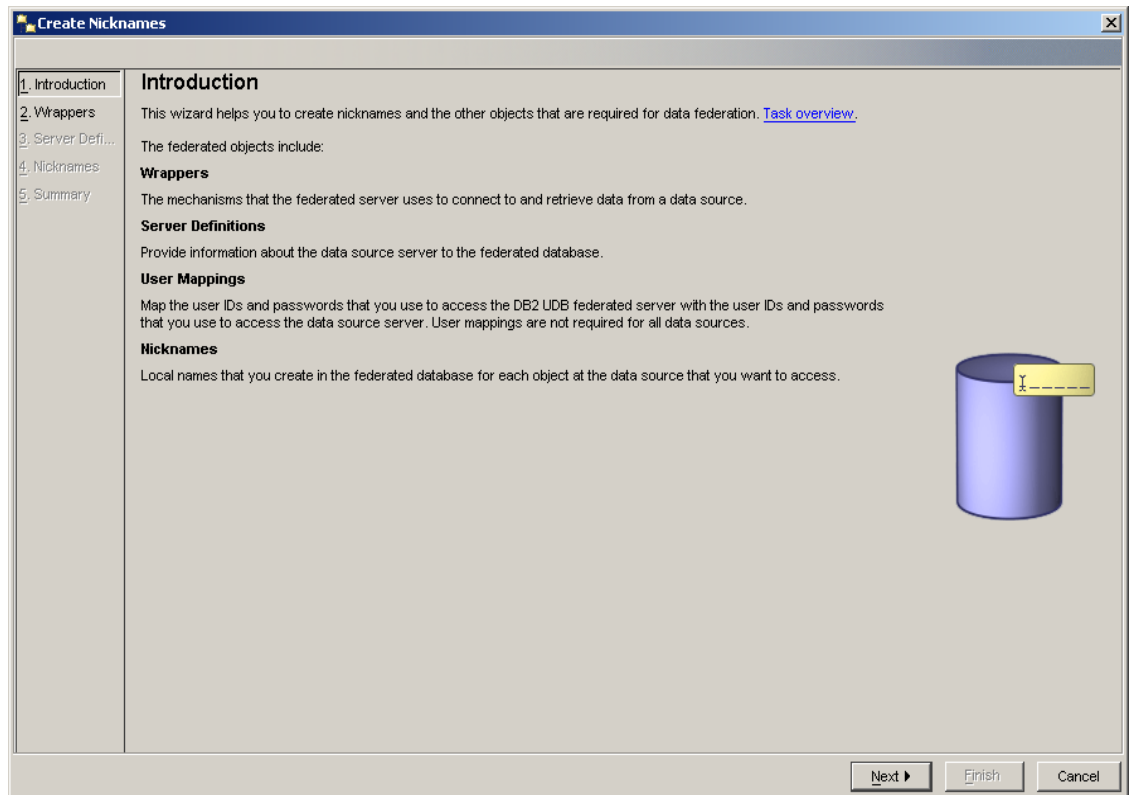

The wizard will walk you through 5 pages to help create the federated objects.

- The first page gives you an overview of the tasks to be performed to setup federated access to heterogeneous data sources.

- The second page will guide you to create a wrapper

- The third page will guide you to create the required server definitions

- The fourth page will guide to create the required nicknames

- The fifth page provides a summary of the tasks that have been performed.

IBM TechWorks

Step 1: Create Wrapper

A wrapper is the adapter that allows DB2 UDB to work with a specific data source. Before any remote servers or nicknames to remote tables can be created, you must create a wrapper to access a specific type of data. In DB2 9, relational wrappers are installed as part of the WebSphere Federation Server install. However, to have the Informix wrapper installed, you must have specified during the custom install of DB2 9 that you want Informix data source support (this item is located under Server support in the install).

__1.   On the first page of the Create federated Objects configuration wizard, click **Next** to go to the **wrapper definition page** (second page), where you will create the wrapper; the wrapper provides federated access to a specifc data source type.

__2.   Use the drop down list to select **Informix** as the **data source type** and click **Create...**

__3.   This will launch the **"Create Wrapper"** window with the values for the fields 'data source' and 'Library name' automatically filled with **Informix** and **db2informix.dll** respectively. Specify **INFORMIX_WRAPPER** as the name for the wrapper.

__4.   Click **OK**.
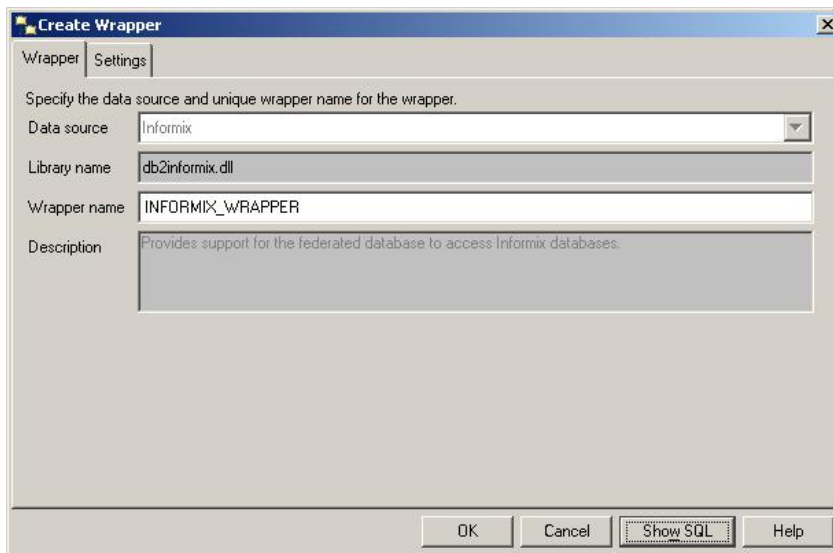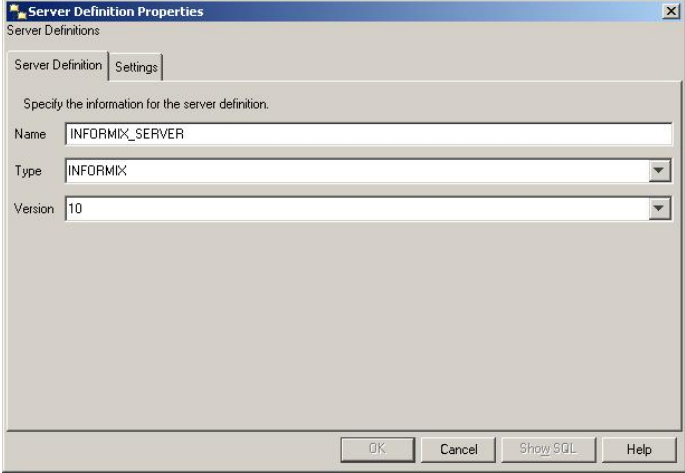
This will close the **"Create Wrapper"** window and return you back to the Wrapper definition page (second page) of the "Create Federated Objects" configuration wizard. **INFORMIX_WRAPPER** will now appear in the list of the available wrappers.

Step 2: Create Server

A server is a specific instance of a remote data source. So for example, if you have an Informix server running in department A, and another instance running in department B, you will have 2 different servers using the same wrapper, INFORMIX_W.

__5.   Select **INFORMIX_WRAPPER** in the wrappers table on the Wrapper definition page (Note: If there is only one wrapper in the table, it will be automatically selected)

__6.   Click **Next** to go to the **Server definition page** of the nickname configuration wizard.

__7.   Since no servers have yet been created using the wrapper **INFORMIX_WRAPPER**, the existing server definitions table is empty. Click on Create... to open the **"Create Server Definitions"** window.

__8.   WebSphere II can discover remote data sources that have been configured on the system. Click **Discover**, to let WebSphere Federation Server discover the Informix data sources that have been configured to access from the lab workstation.

__9.   The discovery function will retrieve one entry **OL_PISTEW**. This is the nodename of the Informix Server used at installation of the product.

__10.   Once you have selected OL_PISTEW, click on **Properties** to open **Server Definition Properties** page. This page allows you to customize the Server definition. Customize the server definition as follows:

   ___a.   **NAME:** Change to **INFORMIX_SERVER**.

   ___b.   **TYPE:** The pull down will show the single choice of **INFORMIX** as we are creating a server definition for an Informix data source using the wrapper created in step 1.

   ___c.   **VERSION:** The pull down shows multiple available versions: from version 7 to version 10.. Since the database **ORDERS** resides on a version 10 instance, we will choose 10 here.

__11. Click on the **Settings** tab, and you will notice that two options have already been selected. Now specify the following:

DBNAME: Type in **ORDERS**.   This is the name of the database in the Informix database containing the tables we are interested in.

NODE: The value for this option is pre-filled as **ol_pistew** as that is the node name of the Informix server we are working with

PASSWORD: Check this option to specify that passwords are sent down to the remote source for validation.

INFORMIX_DB_LOCALE: type in EN_US.8859-1, which is the default locale for Informix database at installation.
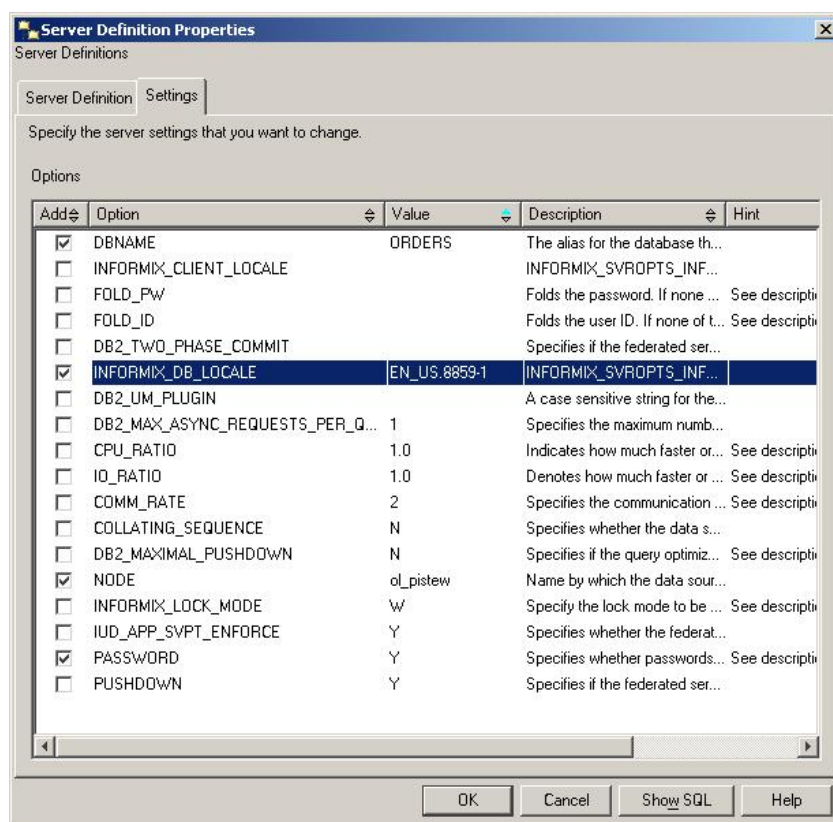
__12. Before clicking **OK** to run the command, you can click **Show SQL**, you should see the SQL:

```
CREATE SERVER INFORMIX_SERVER TYPE INFORMIX VERSION '10' WRAPPER
INFORMIX_WRAPPER OPTIONS (ADD NODE 'ol_pistew', DBNAME 'ORDERS',
INFORMIX_DB_LOCALE 'EN_US.8859-1', PASSWORD 'Y');
```

You can copy this SQL and save it to a file for future.   For this lab, it is not necessary.

__13. Now click **OK** to run the SQL.

Step 3: Create User Mapping

Now that we have the erver defined in the federated database, we need to tell DB2 the userId/pwd to use in order for DB2 to connect to Informix on behalf of the DB2 user. The user mapping allows the user to specify a specific Informix id/pwd for each specific DB2 UDB userId.

__14. Select **INFORMIX_SERVER** in the existing server definitions table on the Server definition page (Note: If there is only one server in the table, it will be automatically selected)

__15. Click **Next** to go to the User Mappings definition page of the nickname configuration wizard.

__16. Since no user mappings have yet been defined for this server, the existing User Mappings definitions table is empty. Click on **Create...** to open the **"Create User Mappings"** window.

__17. In the **"Create User Mappings"** window, click on **ADMINISTRATOR**, which is your current id, then click > to select it. Click the **Settings** tab, type in **informix** for Remote_AUTHID and **ibmpassw0rd** for REMOTE_PASSWORD, both lower cases. Click **OK**.
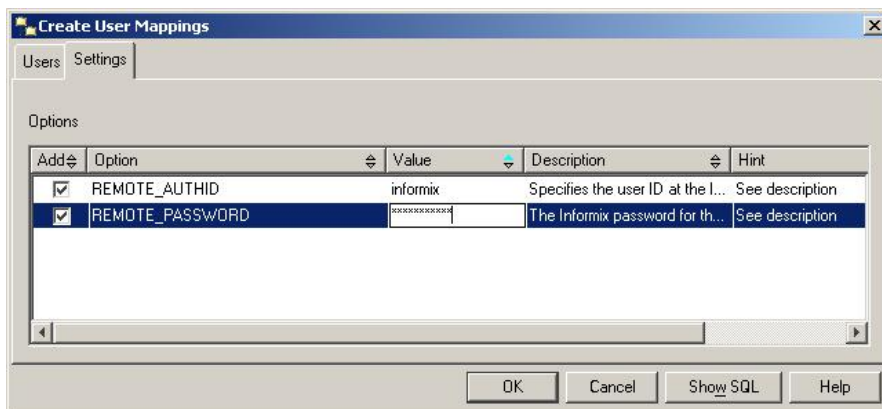
Step 4: Test connectivity

Before proceeding with creating nicknames to the Informix tables, it is a good idea to verify the setup so far is correct. The best way to test this is to access Informix in PASSTHRU mode. When in PASSTHRU mode, DB2 UDB will simply pass the SQL directly to the remote server. This way you can verify whether your connectivity to the server is indeed working.

__18.	Open a DB2 command window: Click **Start →All Programs → IBM DB2 → Command Line Tools → Command Window**.

__19.	In the command window, type

DB2 connect to whs

DB2 set passthru informix_server

DB2 select * from orderitem

>Note the SELECT statement is now passed directly to the Informix database. If your Informix Server is set up correctly, you should see records returned from Informix Server. Leave the DB2 command window open, we'll use it later for testing.

___d.	DB2 set passthru reset

This command gets you out of the passthru mode.

Step 5: Create nicknames

The key to DB2 federated technology is the use of nicknames. A nickname is an object in a DB2 database that behaves exactly like a table, but it is actually a reference to a remote table residing in a specified remote server. This way the full power of SQL is available to nicknames. You can now, for example, join different nicknames/tables and get results from heterogeneous data sources.

__20.	Click **Next** to go to the Nickname definition page of the nickname configuration wizard.

__21.	If you know the remote schema (sometimes called owner) and remote table name, you can enter them directly after left-clicking on the **Add** button. For this exercise, we will use the Discover button to list the remote tables.

Click on the **Discover** button to display the filter window, **"Discover – INFORMIX_SERVER"**.

__22.   Since the tables we are interested in have a name starting with the letters ord, use the filter capability to limit the number of remote table names retrieved. To do this, check the **Remote table name box**, select the Operator LIKE, and type **ord%** in the values field for remote table name and click **OK**.
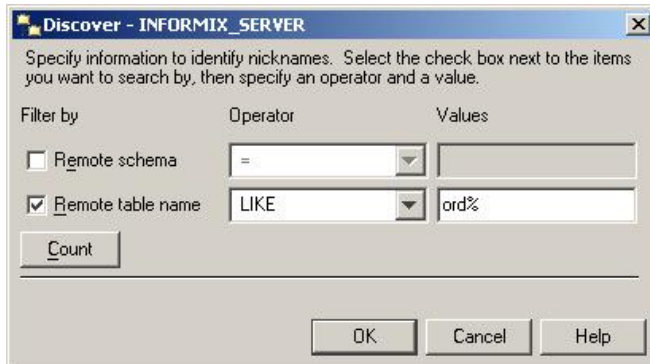
__23.   Remote tables **orderitem** and **ordermain**, discovered from the **ORDER** database are returned to the nicknames table.

__24.   Select the orderitem table, click on the **Properties** button, and write **ORDER_IFX** as the nickname schema. We'd like to make every nickname created in the **ORDER** database under the **ORDER_IFX** schema name. Make similar change to the ordermain table, and specify ORDER_IFX as nickname schema. It is a good practice to put as schema name a mnemonic for the database that you are accessing.

Note that you can select all the remote tables and change at once the schema of their corresponding nicknames by clicking the **schema** button.
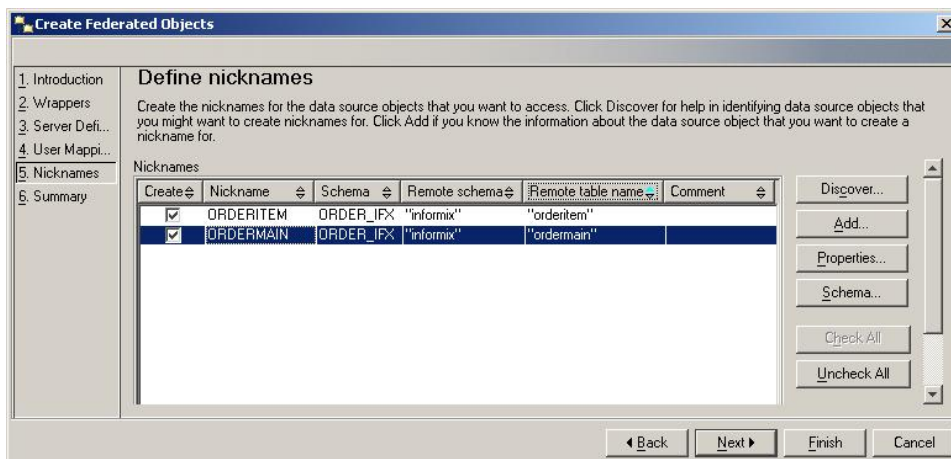
__25.   Click **Next** to go to the summary page and review the nicknames being created. You can also review the SQL statements executed to create the nicknames by clicking **Show SQL** button. Click **Finish** to create the nicknames.

__26.   To see the list of nicknames created, double click on **Nicknames** in the left panel, you should see ORDERITEM and ORDERMAIN (both with a local schema of ORDER_IFX) appear as nicknames in the right panel. (Hint, you may have to refresh before the nicknames appear in the Control Center, or in some occasions need to shut down the

DB2 UDB Control Center and bring it back up if refresh still does not work). To verify the nicknames are working, you can right click on a nickname and select **open** as the popup action.

__27. The **Action Output** window will display a history of past run actions, as well as the current status of running actions. The action output window will allow the DBA to view the input scripts for each entry. The user will be able to select one or more of these completed actions, combine the scripts for each action, and copy the combined scripts to the Command Editor where the input scripts can be edited, executed again, and saved. The Action Output window only shows actions run via the Control Center.

For each new Control Center session, the action output window will be launched automatically the first time an action output enabled command is executed. When the action output window is explicitly closed by the user, it will not automatically re-open when the next action output enabled command is run.

To launch the Output Action window, right-click menu option, **Show Actions** on the Federated Database Objects folder.
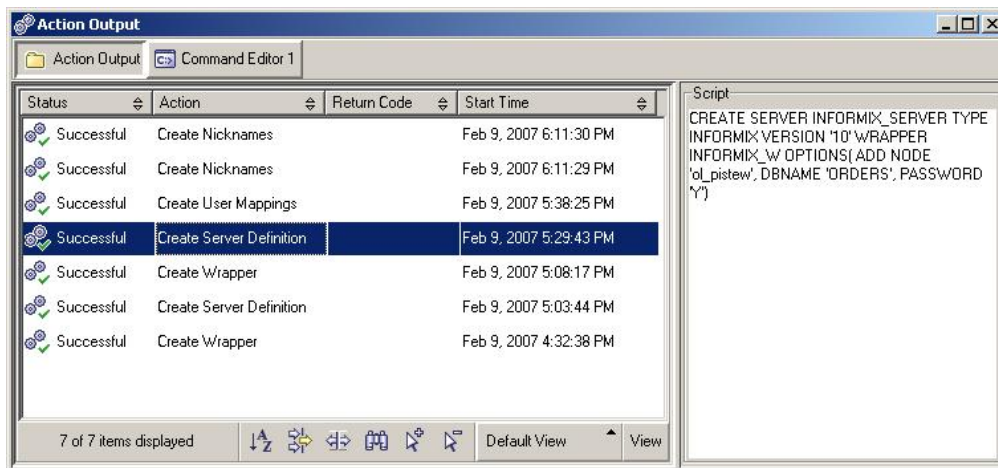
__28. Left-click the ☒ in the upper right hand corner of Output window to close the window.

Step 6: Test scenario 1

Now that we have set up federated access to the data in the ORDER database, all the Informix tables for which we have created nicknames will appear in the nicknames folder of the WHS database under the schema ORDER_IFX; and more important, they can be used as if they are normal DB2 tables. We can now for example do the following from the DB2 command window:

__29.   Find all the order dates and items ordered from the customer with customer number 5. This is a customer with recent orders (i.e., the orders are stored in the operational DB).

- db2 set current schema order_ifx

- db2 select orderdate, ordernumber, totalprice from ordermain where customernumber = 5

You should get back several records.

__30.   Now, let us build a query which finds all the orders that have been placed by customer with customer number 5: this means that we must get the data from both the operational database (ORDER) and the data warehouse (WHS).

Here is the query (you can find this query in file Step3_AllOrdersQuery.sql in directory C:\SOAdemo\Scripts\Scenario2):

```
SELECT CT.FACTS.ORD_NUM as ORD_NUM,   CT.FACTS.CUST_NUM as
CUST_NUM,
      CT.FACTS.PROD_NUM as PROD_NUM,  CT.FACTS.QTY as QTY,
      CT.FACTS.ORD_DATE as DATE,   CT.ORDER_DIM.ORD_STATUS as
STATUS
FROM CT.FACTS, CT.ORDER_DIM
WHERE CT.FACTS.ORD_NUM = CT.ORDER_DIM.ORD_NUM   AND
CT.FACTS.CUST_NUM = 5
UNION ALL
SELECT ORDER_IFX.ORDERMAIN.ORDERNUMBER as ORD_NUM,
      ORDER_IFX.ORDERMAIN.CUSTOMERNUMBER as CUST_NUM,
      ORDER_IFX.ORDERITEM.PRODUCTNUMBER as PROD_NUM,
      ORDER_IFX.ORDERITEM.QUANTITY as QTY,
      ORDER_IFX.ORDERMAIN.ORDERDATE as DATE,
      ORDER_IFX.ORDERMAIN.ORDERSTATUS as STATUS
FROM ORDER_IFX.ORDERMAIN, ORDER_IFX.ORDERITEM
WHERE ORDER_IFX.ORDERMAIN.ORDERNUMBER =
      ORDER_IFX.ORDERITEM.ORDERNUMBER
      AND ORDER_IFX.ORDERMAIN.CUSTOMERNUMBER = 5;
```

You can execute the above query by issuing from the DB2 command line:

```
DB2 connect to WHS
DB2 -tvf  C:\SOAdemo\Scripts\Scenario2\Step3_AllOrdersQuery.sql
```

Notice that we have more records than what was returned in the previous query. Data has been collected from both databases.

__31.   We will now create a view ADMINISTRATOR.ALLORDERS to get all orders of all customers. The definition of the view is:

```
CREATE VIEW ADMINISTRATOR.ALLORDERS
     ( ORDERNUMBER,CUSTOMERNUMBER,  PRODUCT_NUMBER, QUANTITY,
     ORDERDATE, ORDERSTATUS) as
SELECT CT.FACTS.ORD_NUM,   CT.FACTS.CUST_NUM,
     CT.FACTS.PROD_NUM,  CT.FACTS.QTY,
     CT.FACTS.ORD_DATE,   CT.ORDER_DIM.ORD_STATUS
   FROM CT.FACTS, CT.ORDER_DIM
   WHERE CT.FACTS.ORD_NUM = CT.ORDER_DIM.ORD_NUM
UNION ALL
SELECT ORDER_IFX.ORDERMAIN.ORDERNUMBER,
     ORDER_IFX.ORDERMAIN.CUSTOMERNUMBER,
     ORDER_IFX.ORDERITEM.PRODUCTNUMBER,
     ORDER_IFX.ORDERITEM.QUANTITY,
     ORDER_IFX.ORDERMAIN.ORDERDATE,
     ORDER_IFX.ORDERMAIN.ORDERSTATUS
   FROM ORDER_IFX.ORDERMAIN, ORDER_IFX.ORDERITEM
   WHERE ORDER_IFX.ORDERMAIN.ORDERNUMBER =
     ORDER_IFX.ORDERITEM.ORDERNUMBER ;
```

__32.   Create this view by issuing from the DB2 Command line:

```
DB2 -tvf  C:\SOAdemo\Scripts\Scenario2\
Step4_CreateView_AllOrders.sql
```

__33.   To find out all the orders placed by a specific customer (5), issue now the query:

```
DB2 select * from ADMINISTRATOR.ALLORDERS where customernumber=
5
```

This is a federated query that selects data from a view built as a union of tables from the warehouse and nicknames on tables in the ORDER database.

We will now show you how to expose this query as a service.

Part 2 – Use the Information Server Console

- to deploy a federated query as a service

The Console for IBM Information Server is the tool which lets you deploy as a service a DB2 federated query. A wizard guides you through the steps needed to perform this deployment.

The wizard will walk you through a number of pages to help you create the service; the tasks that will be performed during the creation of the service are:

1. name and describe the new service.

2. choose one service interface binding (e.g., SOAP over HTTP, EJB, …)

3. define the operation that the federated service will invoke: in example, it will be an SQL statement; it could also be a stored procedure

4. set up the request and response messages for the operation

5. set run time parameters.

Before starting the Console for IBM Information Server, you must have an application server up and running. In our configuration, we are using WebSphere Application Server (which was installed as part of the IBM Information Server install). Verify that the IBM WebSphere Application Server service is already started (the service name is IBM WebSphere Application Server V6 – pistweNode02); if it is not started, start the service.

__34. Open the Console for IBM Information Server (**Start → All Programs → IBM Information Server → IBM Information Server Console**)

When you are prompted for user name and password, enter:

- user name: **administrator**

- password: **ibmpassw0rd**
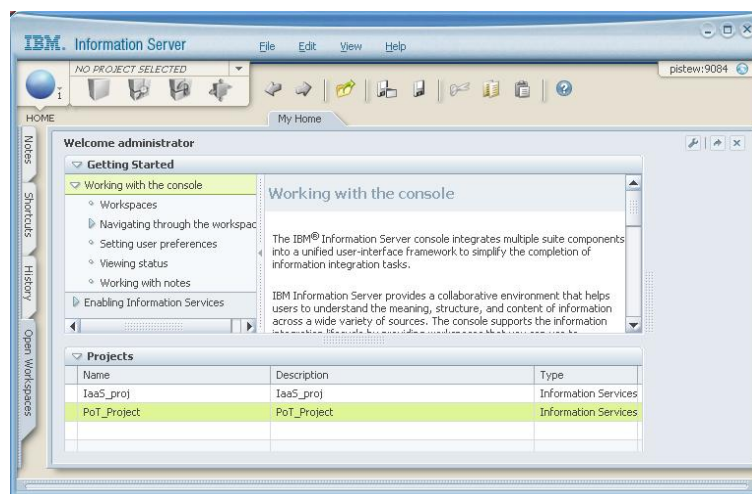
__35.   Open the PoT_Project that we defined during the lab setup.

Select **File → Open Project**. In the Open Project window, select PoT_Project, and click the Open button.

__36.   Open the Information Services Application, PoT_Application.

Applications in WISD mark the deployment unit. You can include any number of services within an application. During the lab setup, we created one application, PoT_Application, within the PoT_Project.

Open the Information Services Application window
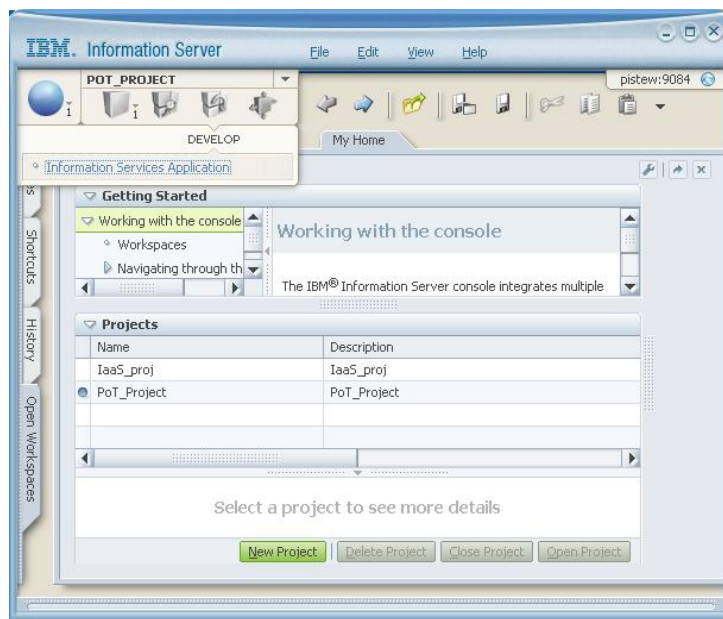
___e.   Click on the **Develop** icon, and select **Information Services Application** from the menu.

___f.   This opens the Information Services Applications window; it displays the PoT_Application we already created.
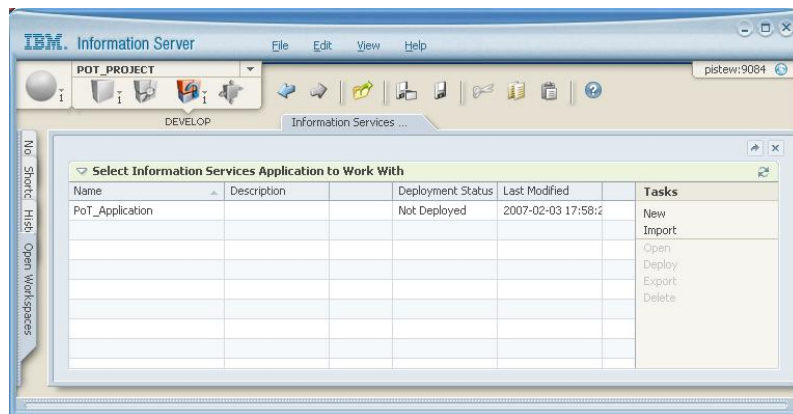
___g.   Open the PoT_Application: select the application and click on **Open**.

___h.   The next window is the environment from which you can create and deploy
          services within this application.
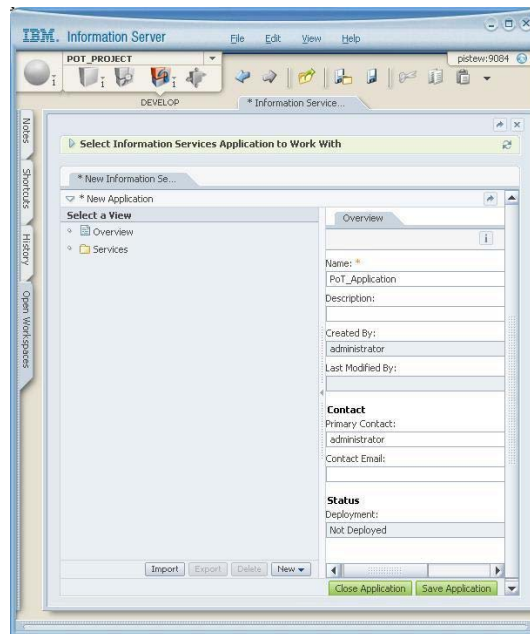
__37.   Create an Information Service for a federated SQL statement

          From the PoT_Application window, click the **New** button. Within the New menu, select
                  Service.

IBM *TechWorks*

The wizard to help you create and deploy the new service is being opened.
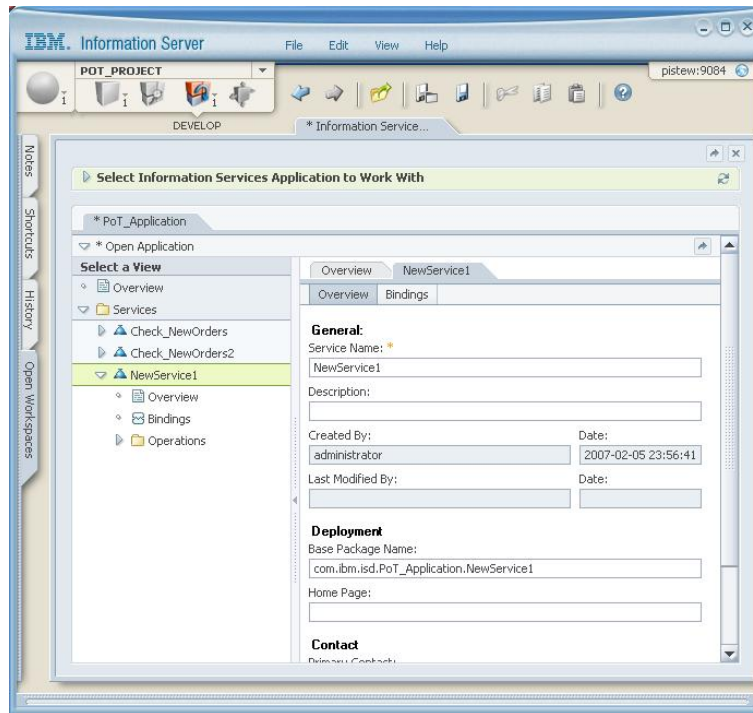
___i.   On the **Overview** page of the Service wizard, enter the following information:

• Service Name: **OrdersList**

• Base Package Name: **com.ibm.isd.PoT_Application.OrdersList**
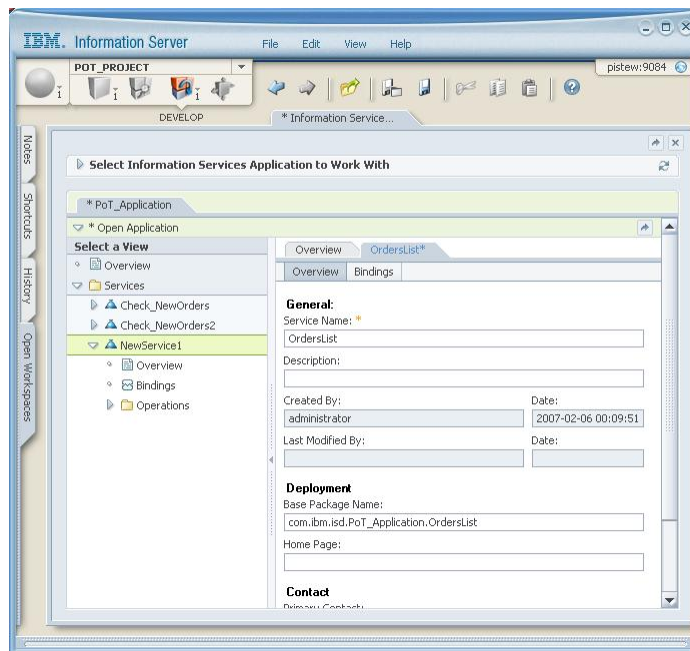
___j.   On the Bindings page, click on the **Attach Bindings** menu button, and select **SOAP Over HTTP** as binding.
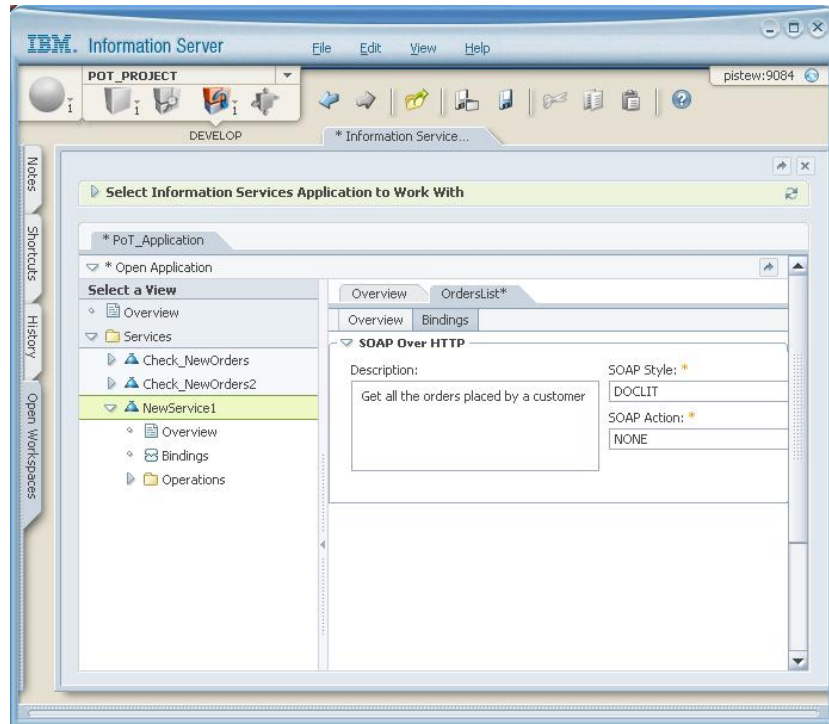
___k.   If you want, you can also enter a description of the service and binding.

___l.   You will now specify the operation performed by the service. Notice that creating a new service also creates a new empty operation: expand the Operations folder, and double-click on newOperation1; change the name of the new operation to **getOrdersList**.
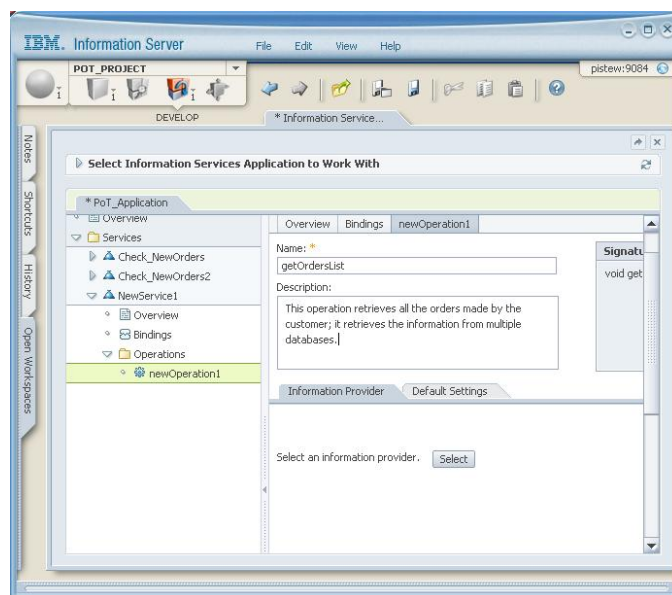
___m. This operation is implemented using the DB2 Federation Server Information Provider: click on the **Select** button to choose the Information Provider, and pick the item **DB2 or Federation Server**.

For subtype, select **SQL Statement**; note that you could invoke also a Stored Procedure to perform the operation.

For Action, select **Browse Databases**

For Federated Object, select the database **WHS**

Open the WHS database folder, and select **Views**

Select the View **ADMINISTRATOR.ALLORDERS** in the list of Views.
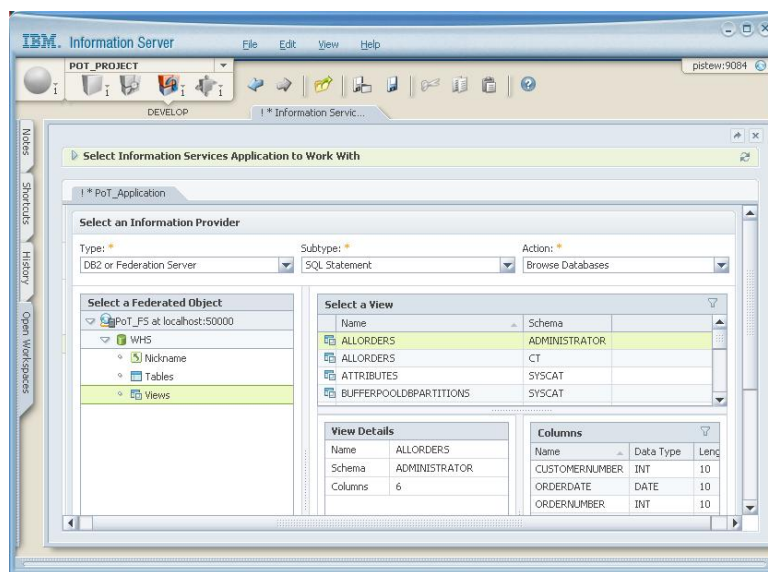
Click **OK** at the bottom of the window.

___n.   The detail of the service operation is displayed; the details include the inputs, the outputs, the SQL statement and the information provider properties.
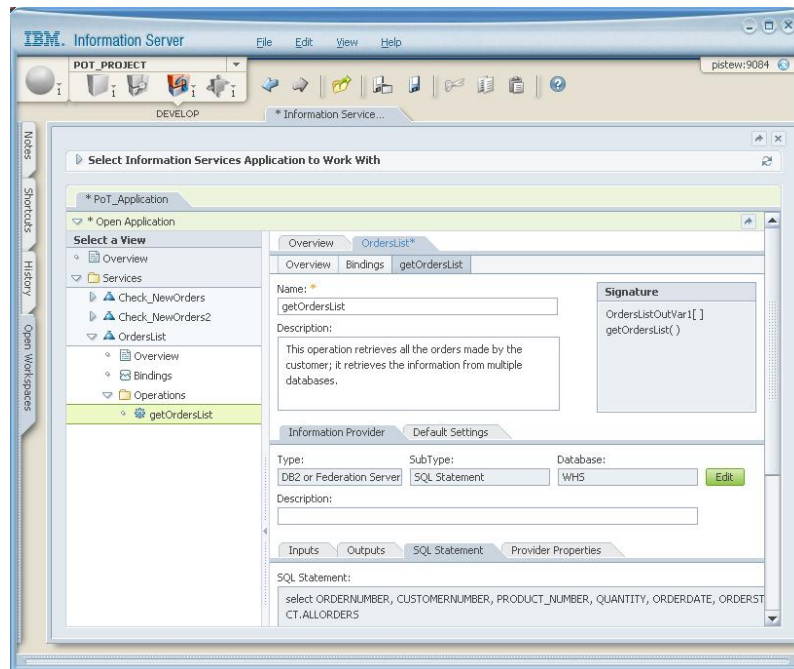
___o.   We will make a small modification to the SQL statement; to modify the statement, open the SQL statement tab, click on the **Edit** button next to Database, and add at the end of the statement the following where clause: **where customernumber = ?.**

___p.   If the statement to build was much more complex, you could use the SQL Builder to help you formulate the statement.

___q.   Click **OK**.

____r.   Notice that Inputs tab now contains an argument
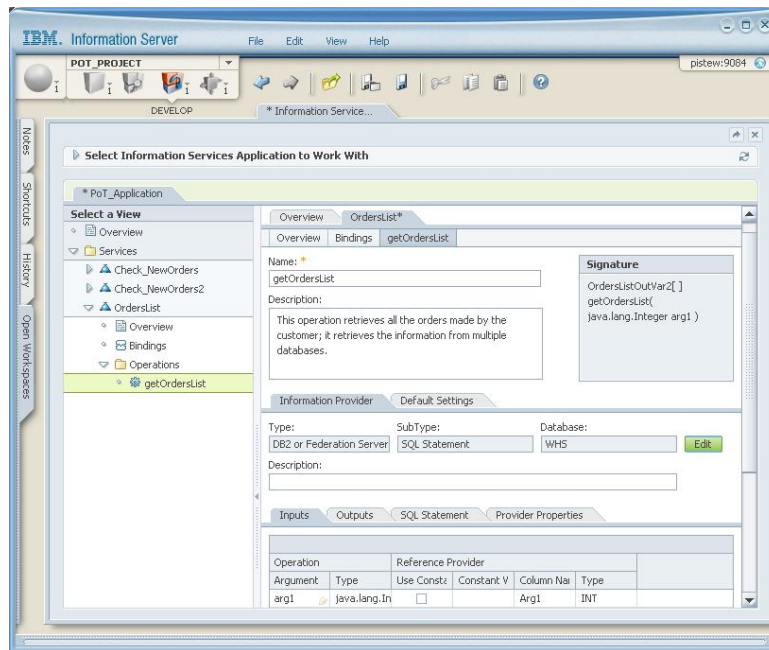
____s.   Replace the value of arg1 under Argument with **customernumber**.

____t.   The query we have just built retrieves all the orders made by a customer, for which we provide the customernumber as input. You can browse through the tabs to check once again the inputs,outputs, and SQL statement. Note that the view we have selected is the view we defined in DB2 in the first part of this exercise.

____u.   Click **Save Application**, then **Close** the **Application**
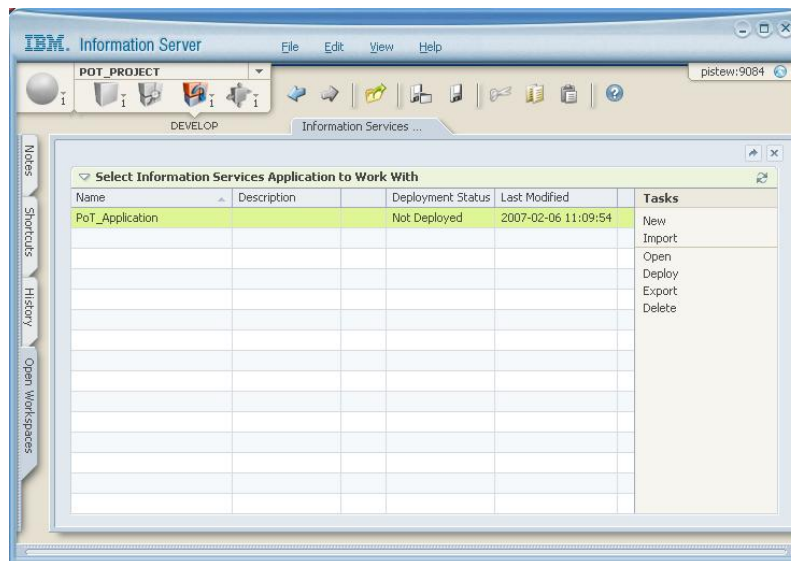
__38.   Deploy the application

Our application, PoT_Application, contains now 2 services; in the first scenario, we already deployed this application with the first service we built. You will not be able to deploy an application with the same name if it has already been deployed. Just try, by selecting the Deploy task button, and you will get an error message.

In order to successfully re-deploy an application, you must first undeploy the application.

To un-deploy an application, select the Operate icon on the Navigation bar, and select Deployed Information Services Applications from the menu.
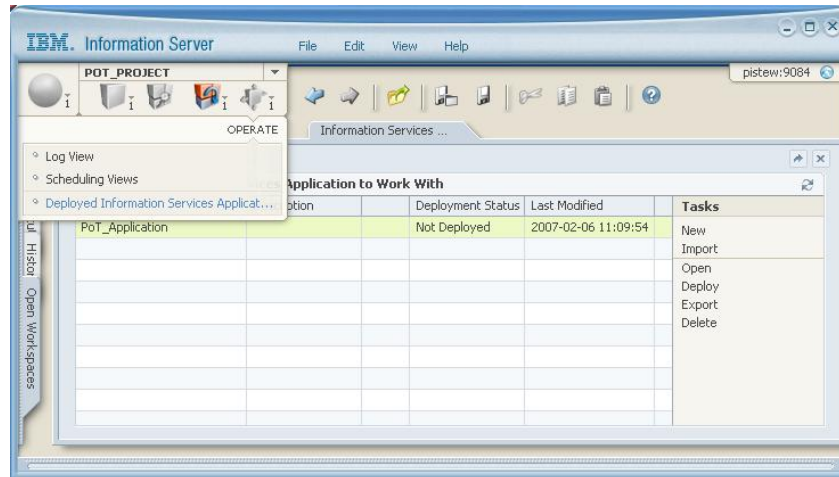
___v. Select the PoT_Application, and click Un-deploy.
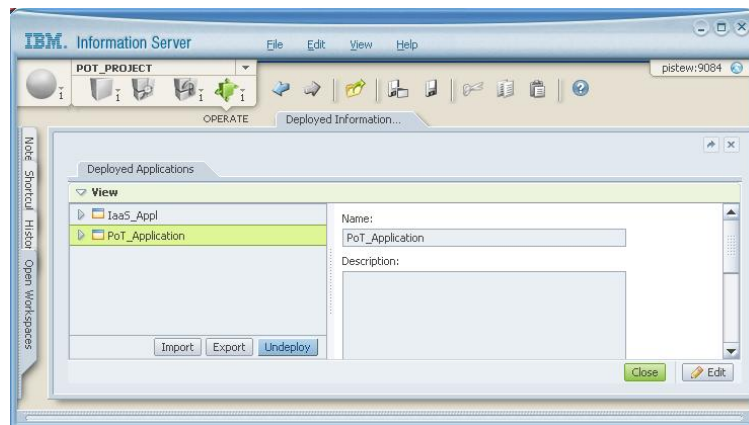
___w. Click OK when the undeployment completes.

___x. Now select the Develop icon in the Navigation, and select Information Services Application from the menu.

___y. In the window displaying the Information Services Applications, select the Application named PoT_Application .
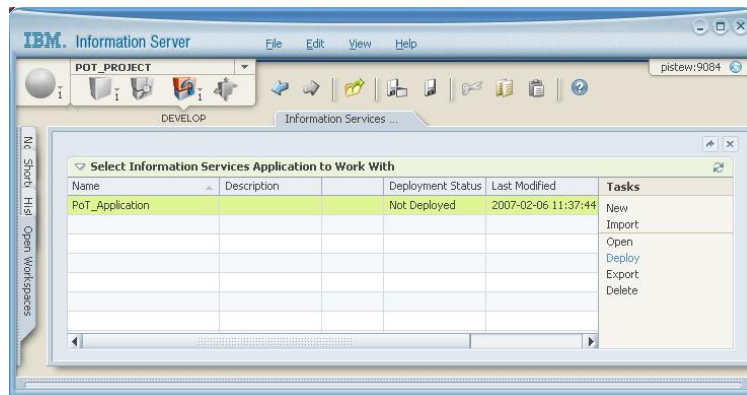
___z. Click **Deploy.**

___aa. The window with the Service Objects to deploy is displayed; deploy the services named CheckNewOrders2 and OrdersList; uncheck the service named Check_NewOrders
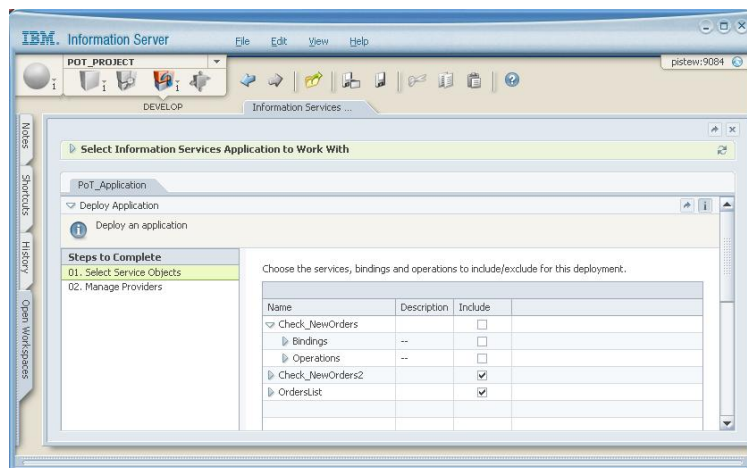
___bb. Select the Manage Providers item in the Steps to Complete.

___cc.   Select an operation within its respective service to see the detailed information captured about the corresponding information provider.
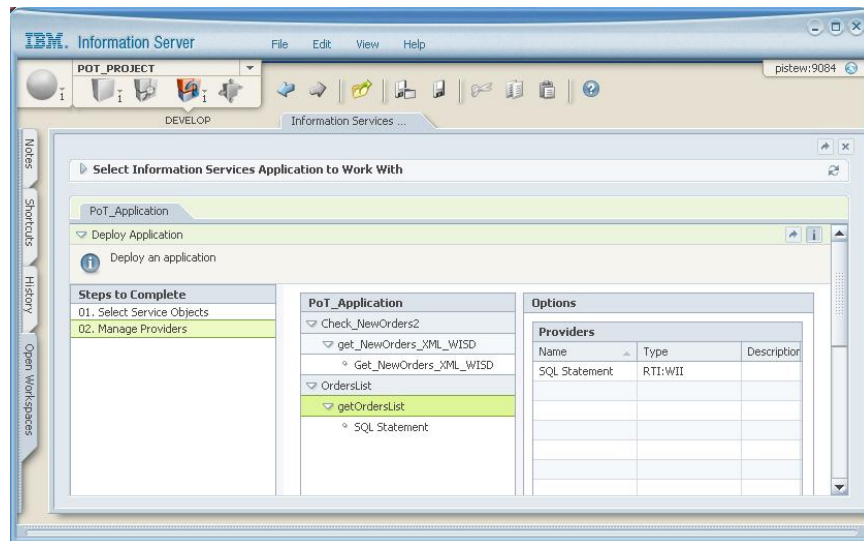
___dd.   To see the input and output, select SQL statement under the operation name.

___ee.   Now, deploy the application: the Deploy button is located at the bottom of the window.

___ff.   While the application is being deployed, you can look at its status by clicking on the **Details** button. When the application is deployed, the status changes from Executing to Completed.
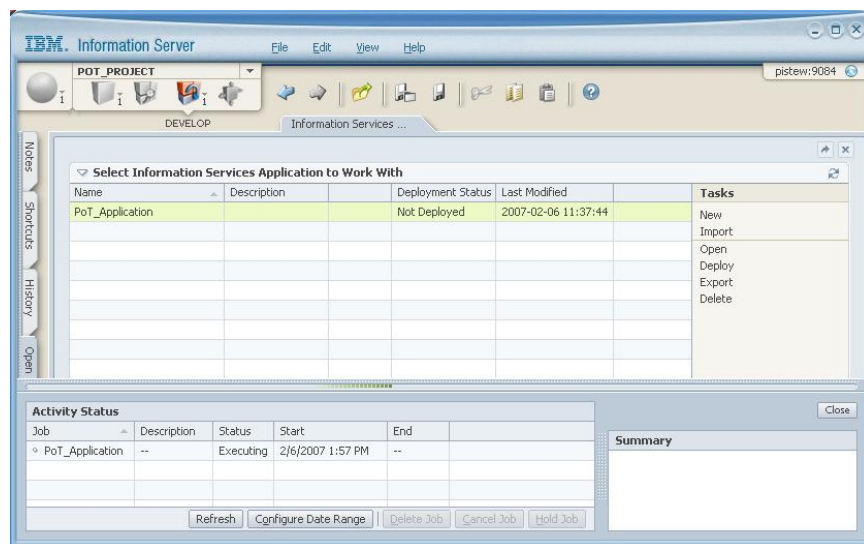
___gg.   Close the Activity Status window. The application is now successfully deployed.

__39.   Examine the WSDL file

To verify the deployment, we will generate the Web service definition language (WSDL) document for our new service. WSDL contains all the necessary descriptions (meta data), that a client application would need, to invoke the service. WISD generates the WSDL "on the fly", which means if your application was not deployed successful, you will not be able to generate the definition. For complete detail, refer to the same section in scenario 1.

___hh.   Open the "Deployed Information Services Application" window: on the WISD Navigation bar, click the OPERATE icon and select *Deployed Information Services Application* from the menu.

___ii.   In the Deployed Applications window, expand the PoT_Application folder.

___jj.   The folder contains now the name of 2 services
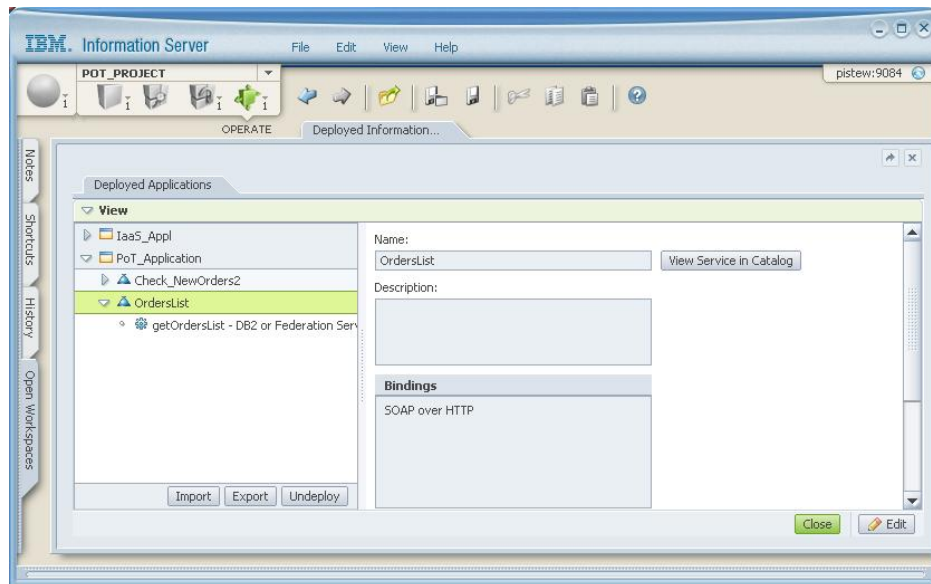
___kk.   Select the **OrdersList** service, and select the **View Service in Catalog** button.

___ll.   The Information Server Administrator Web Client is being opened; it displays the Properties of the service we just selected.
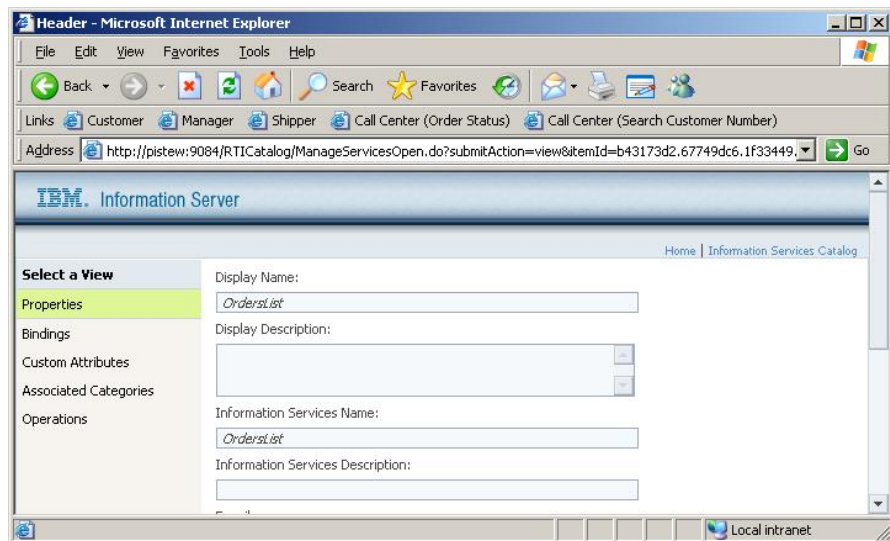
___mm.   Open the **Bindings** view, and expand the **SOAP over HTTP** window; this page gives you access to the WSDL file; from the Bindings page, click on Open WSDL Document. The WSDL file is automatically generated, and displayed in a new browser window.
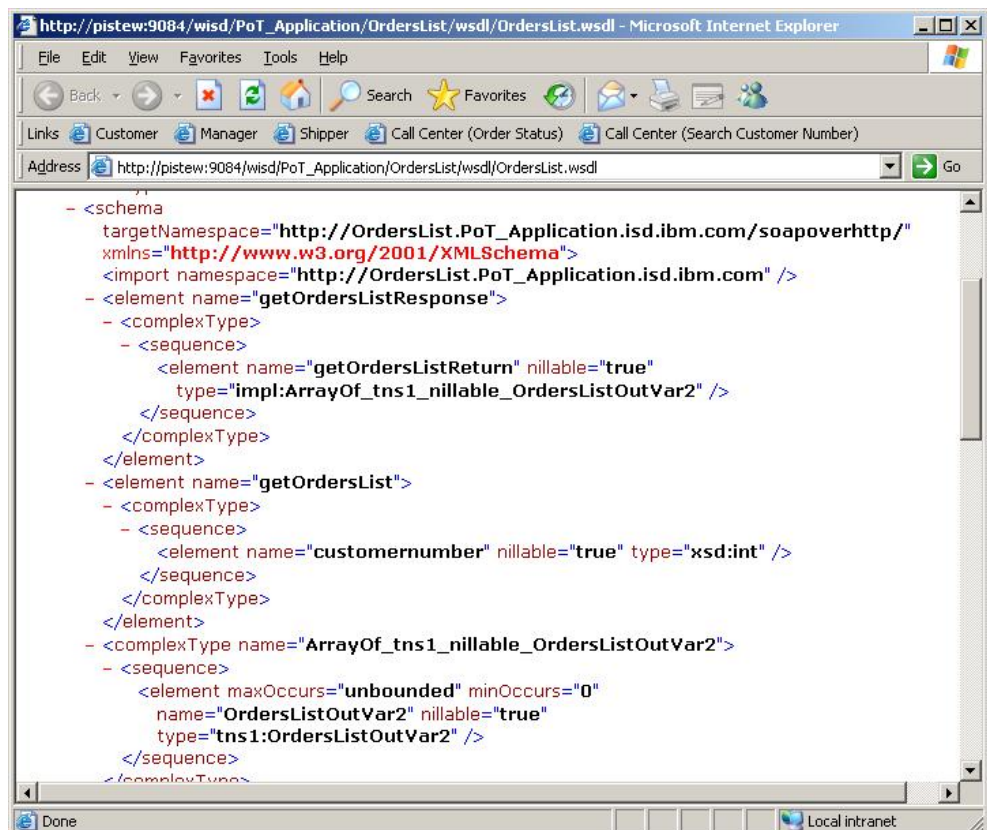
___nn.   Look at the input and output definition contained in this WSDL file.

___oo.   Save the WSDL file in the folder C:\SOADEMO\Results.

___pp.   Keep the name of the URL associated to the WSDL file:

http://pistew:9084/wisd/PoT_Application/OrdersList/wsdl/OrdersList.wsdl

___qq.   Close all the open windows.

___rr.   We are now ready to test the service

__40.   Test the service

To check the service, we will use the WebSphere Integration Developer environment.

Open WebSphere Integration Developer:

**Start → All Programs → IBM WebSphere → Integration Developer v6 →
WebSphere Integration Developer v6**

Accept the workspace as displayed: **SOAiis**.

Select **Run → Launch the Web services** Explorer.

In the Web Browser pane, select the icon representing WSDL Page.
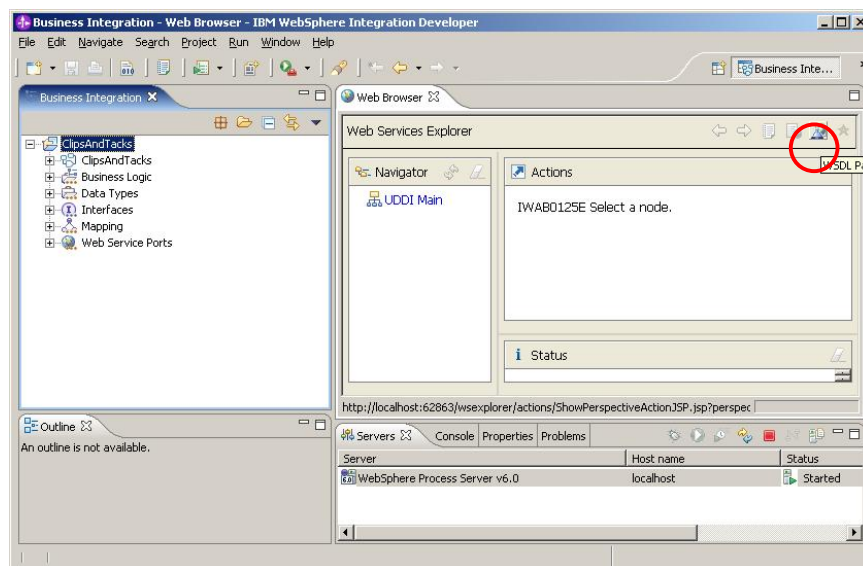
___ss.   Click on WSDL Main in the Navigator

___tt.   Enter the WSDL URL:

http://pistew:9084/wisd/PoT_Application/OrdersList/wsdl/OrdersList.wsdl
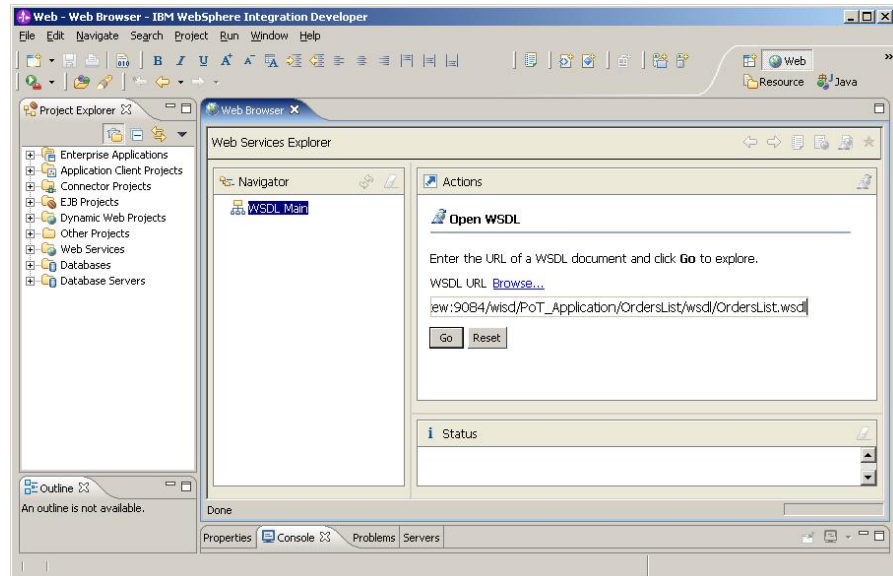
___uu.   Click **Go**, to get the operation name associated to the service.

___vv.   The next screen displays the operation name associated to the service. Click on the operation named **getOrdersList**. This operation takes one input parameter, the customer number; enter the value **5**.
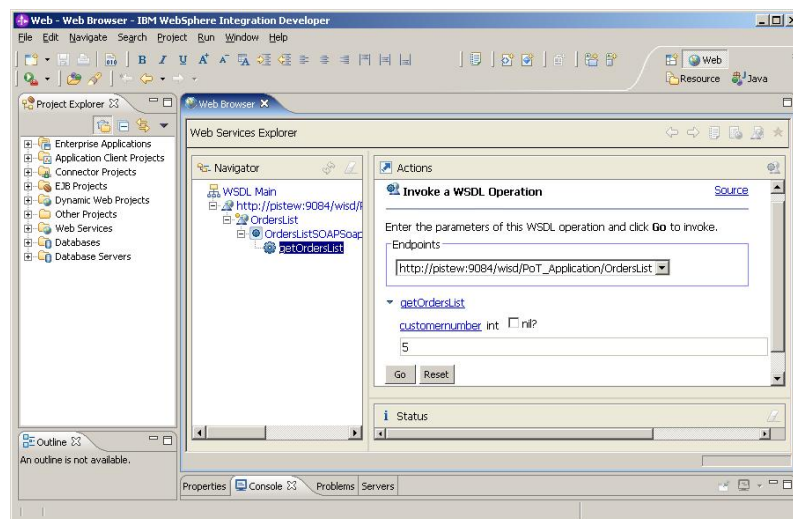
___ww.   Click on **Go**.

___xx.   In the Status window, you receive the XML string with the result
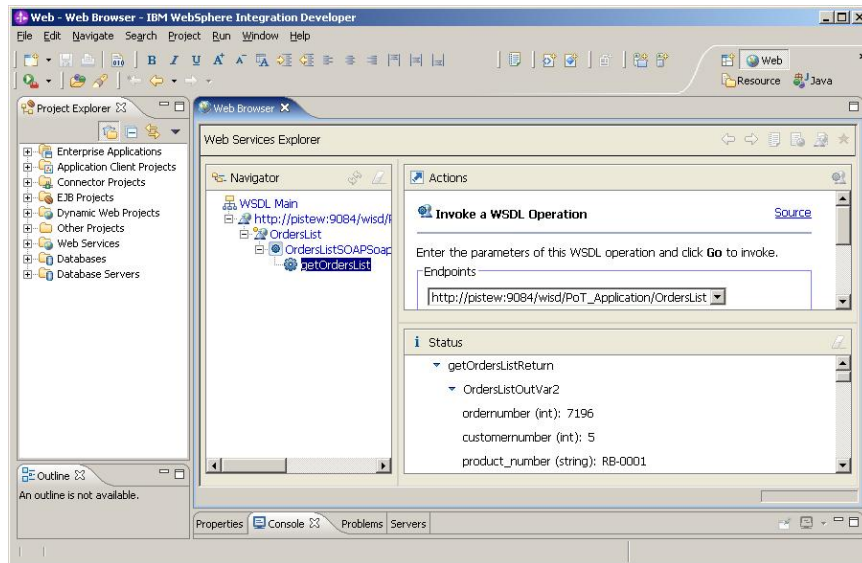
___yy.   You can scroll down the status window, and verify that the data returned contains orders that are stored in the operational database (recent orders), and orders stored in the data warehouse. This service has effectively been using the capabilities of DB2 federation server.

This concludes this exercise; we have used IBM Information Server to expose a federated query as a service.

Summary

In this section, we have shown how to use IBM Information Server to expose a DB2 federated query as a service. Although our query was a federated query (the view used in the query is defined over tables and nicknames), you can expose any valid DB2 query as a service using the WebSphere Information Services Director (WISD) component of IBM Information Server.

The important steps required to expose a DB2 query as a service are:

- Setting up a federated environment, using DB2 Control Center of WebSphere Federation Server. This setup, typically done by an administrator, results in the creation of multiple federated objects, the hierarchy of which is:

  - Wrapper

  - Server – multiple servers may be defined for each wrapper object

  - Nicknames – multiple nicknames may be defined within each server.

- Creation of a view on top of a combination of nicknames and local tables; this makes it easier for applications to just query the view, instead of having to write a rather long and complex query.

- Creation of a service invoking the federated query as its operation; the creation of the service is performed using WebSphere Information Services Director. In order to use Federation Server as an Information Provider, we had first to configure federation server as an information provider; this was done in the lab setup section.

- Deployment of the application containing the service, using WebSphere Information Services Director

- Generation of the WSDL file associated to the service; this task assures that the deployment was successful. This step is performed using Information Server Administrator Web Client.

- Test your new service; various tools can be used. Here, we use the WebSphere Integration Developer environment to test the service, and see the result from invoking the federated query.

Congratulations, you've completed scenario IaaS !

Please wait for the next lecture before proceeding to the next lab.

15 Lab   Monitoring and Business Dashboards Labs

Overview

The labs are designed to provide hands on experience with many of the capabilities available in WebSphere® Business Monitor.

The first four labs use a live WebSphere Business Monitor running on your laptop to provide a view from and end user perspective. End user capabilities can be limited to those appropriate by role/group of the user, but for these labs, users have full access. You will be viewing and modifying business dashboards based on process models in the system.

Lab 15.1 – Dashboards First Look

> This lab includes a review of the online Getting Started pages, then explores several different dashboards in the system. These dashboards contain a mix of the different views available including Key Performance Indicator, Diagram, Instance, Dimension and Alert views.

Lab 15.2 – Create and Personalize Dashboards

> In this lab, you will create a new dashboard and add views to it, as well as explore the personalization options available.

Lab 15.3 – Run live processes and view with Dashboards

> In Labs 1 and 2, you viewed information from processes already completed. In this lab, you will initiate new process instances running on a live WebSphere Process Server, add information to those requests and check the effects on the dashboards in real time.

Lab 15.4 – Create new KPI's; Alert Subscription; Export Actuals

> In this lab, new Key Performance Indicators (KPI's) are defined, and added to a dashboard for viewing. Options for subscribing to alerts and exporting monitored results are examined as well.

The last two labs explore the preparation required for monitoring. These labs may be considered optional depending on your interest area, but they reflect some of the activities required to enable the dashboards you used in the first four labs.

Lab 15.5 – Create Business Measures (Optional)

> WebSphere Business Modeler can optionally be used to define a preliminary version of the measurements used for monitoring. This lab guides you through the steps required to add these Business Measures to an existing process model. A Business Analyst role would typically perform these steps.

Introduction

WebSphere Business Monitor offers two implementations for the end user dashboard. The Portal implementation uses portlets to display the various measurements of interest, and runs in a WebSphere Portal Server environment.

The alternative implementation in a non portal environment is called the Web Dashboard. For the labs today, you will be using the Web Dashboard installation.

For both implementations, the end user views the dashboards using a browser. The content delivered and the functions available in both implementations are the same, the difference in appearance is minor.

To maximize the flexibility and the range of hands on activities for the labs, a running server environment is installed on your laptop containing all the components of WebSphere Business Monitor. This is not supported in a full production environment. Due to the laptop constraints in memory and horsepower, you will notice some delays in response at some points in the labs.

As you go through the labs, consider some of the various Business Measurements that are important in your environment, and give some thought to how they might be displayed in the dashboard views you see. A wide variety of combinations is possible, and the labs are intended to present some reasonable samples, but the business environment, the roles and responsibilities of the intended users is central to configuring effective and insightful views of business performance.

IBM® currently offers pre-packaged dashboards for some industries, and more are being added. Please contact your local Client Team for additional information.

Please take your time, feel free to ask your instructor for assistance, and enjoy the labs.

Lab 15.1 - Dashboards First Look

In this lab you will examine the Getting Started pages for new users of WebSphere Business Monitor and then briefly look at some of the custom dashboards running on this server.

For this Lab, the server has already been started, and you will login to the Web Dashboards using a browser. WebSphere Business Monitor can also be implemented in a WebSphere Portal Server environment. The same functions are delivered whether using Web Dashboards or Portal.

Getting Started pages

In this section you will review the content supplied in the Getting Started pages that come with WebSphere Business Monitor. These pages are intended for new users, or users who need a quick reminder on the functions available to them using WebSphere Business Monitor. The pages include some animated video pointing out specific functions.

__1.   From the desktop, double click the shortcut for the **Business Dashboard**. You will be using the Firefox browser for this lab.
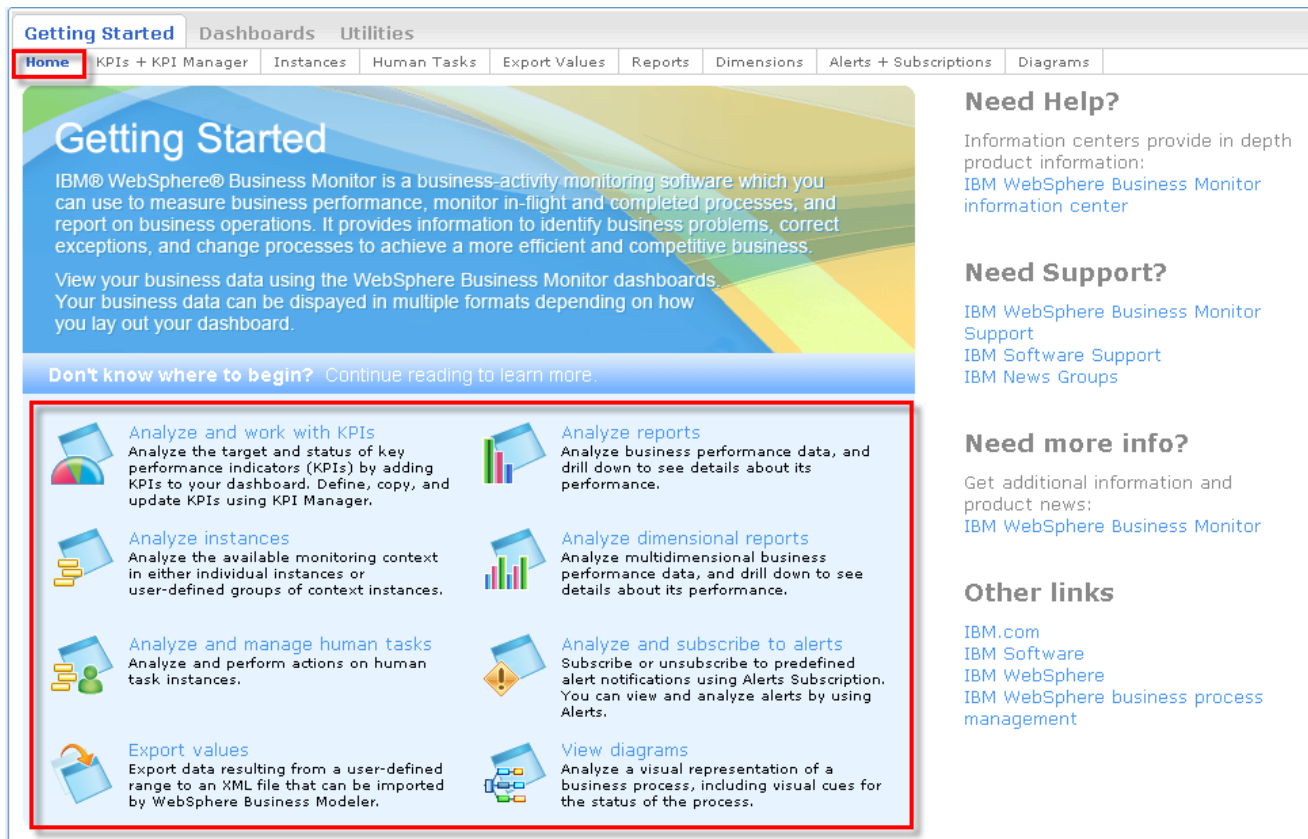
__2.   Due to our lab setup, you will receive a security challenge, just click **OK**.

__3.   You are now prompted to Login to WebSphere Business Monitor. The User ID and Password are pre-filled for you, click **Login**.

__4.   The WebSphere Business Monitor – Getting Started – page opens on the Home tab. In the light blue section below the Getting Started paragraphs, you will see eight topics introduced. On the far right, are links to IBM sites for additional information (these links are disabled in our lab environment).

__5.   The blue paragraph headings for each topic link to pages with more details. These detailed pages are also represented by the tabs across the top of the page, as shown below.
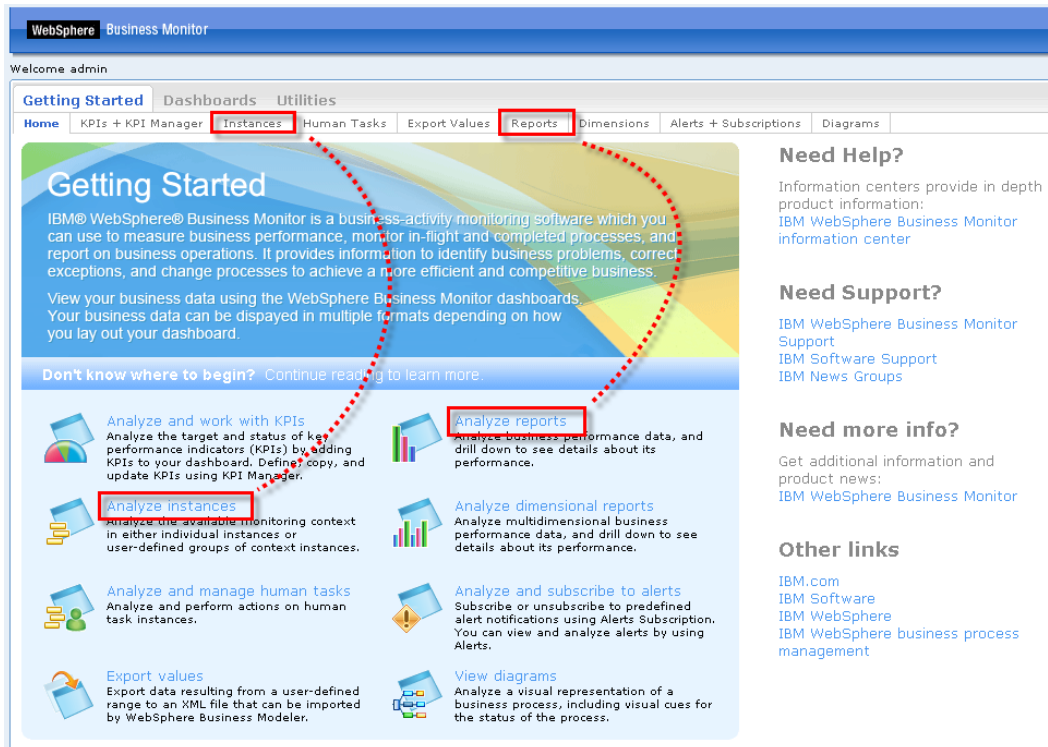
__**6.**   Click the link to learn more about the topic: **Analyze and work with KPIs.**
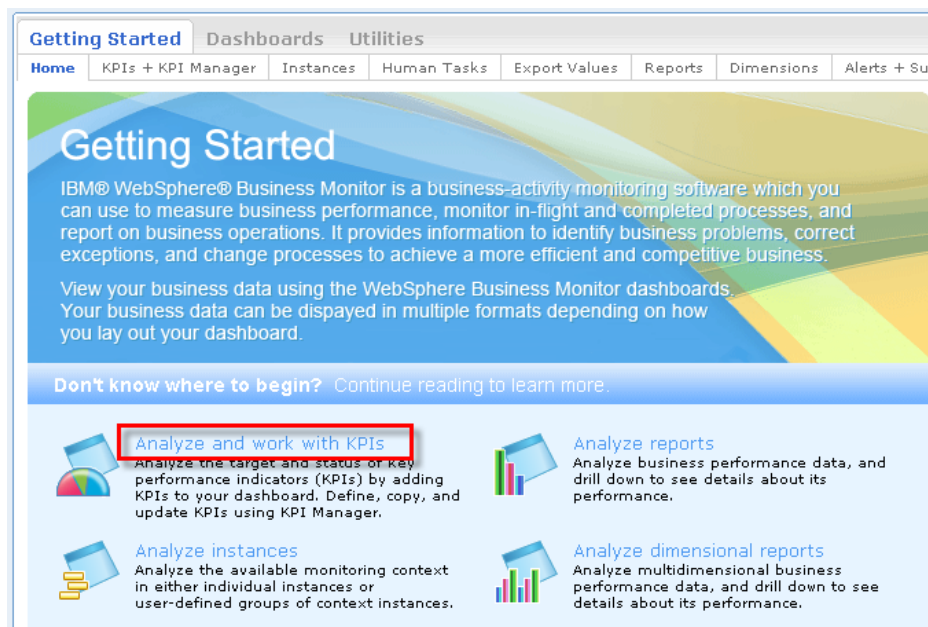
__7.   Click the first link under Key Performance Indicators to **View gauge formats**. You will see Half
        Gauges, Full Gauges, Bar Graphs and Table View formats displayed.
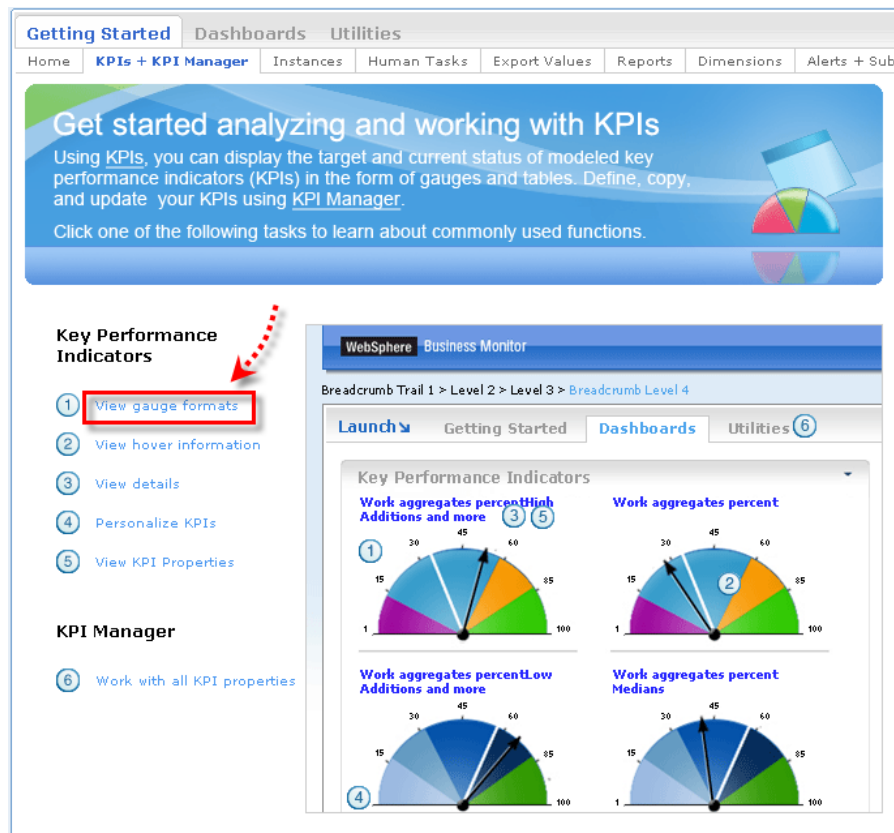
__8.   Click each of the remaining topics 2-6 under Get Started analyzing and working with KPIs, and
        become familiar with some of the techniques used to display the information.

__9.   Click the **Home** tab or use the browser's green '**back**' arrow button to get back to the list of
        topics.
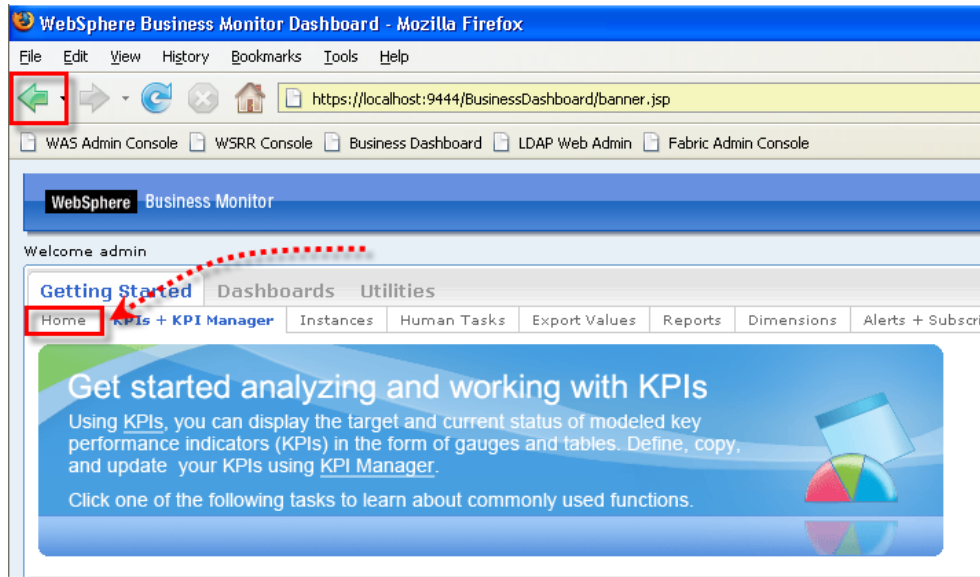
__10. Now examine the remaining topics to become familiar with some of the basic functions available on the WebSphere Business Monitor dashboards. You can either use the tabs across the top or click on the title text in the light blue section; they both link to the same pages.

___xxxviii. For each topic, take the time to watch the short video clips demonstrating the available actions in the various panels. While these clips don't supply much detail, you will begin to get a sense of what capabilities are available. Keep in mind, these same panels would be available to new users in a production environment, to reduce the time to become productive users.
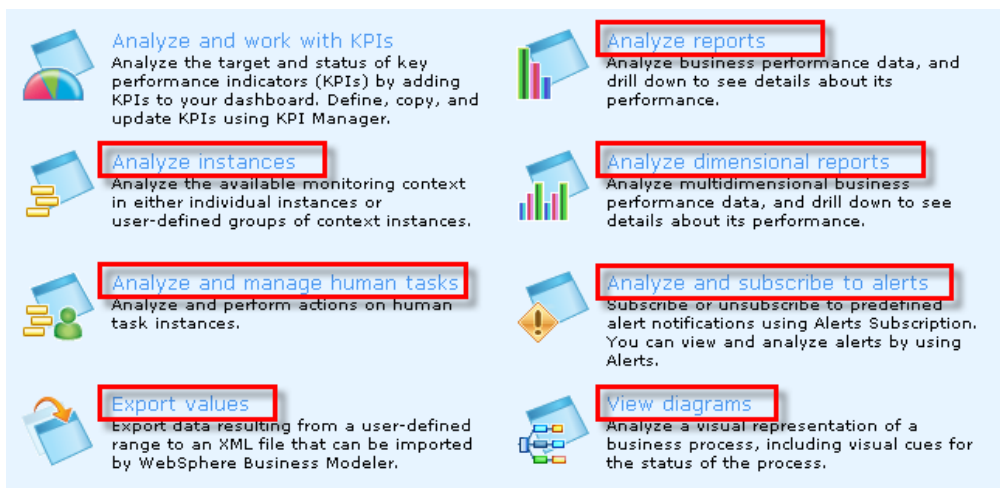
__11.   When you have reviewed all the Getting Started topics in the light blue section, click the **Getting Started > Home** tab.

__12.   You have completed this section.

In this section you will briefly examine existing dashboards running in the system. Dashboards are collections of different views or panels which are configured to show information from different models that have been loaded. Some of the dashboards available on your server show information from one process model; others show data from a different model. In our examples, the dashboards are all related to process models, but WebSphere Business Monitor can also monitor events unrelated to business processes if needed.

__1.   Next to the Getting Started tab, click **Dashboards**.
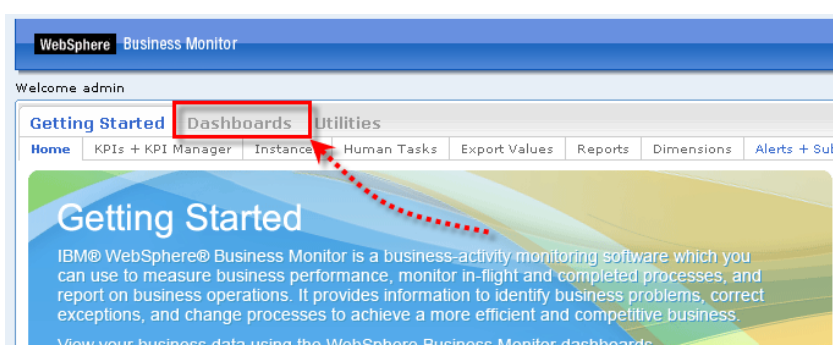
__2.   When the **Dashboards** tab is clicked, it opens to the page that was last visited – in this case the **AV Manager** page. This page was designed with the Account Verification Manager's role in mind. When you click **Dashboards** later on, it will open on the last page you had open.

___xxxix.   If for some reason the **AV Manager** page does not open, you can navigate to it quickly by clicking the tab.
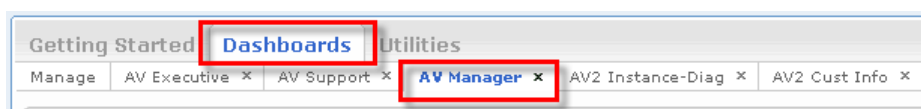
___xl.   This page contains three distinct views:

__a. KPI view

__b. Human Task View

__c. Alert view.

___xli.   Take a moment and review some of the data presented.
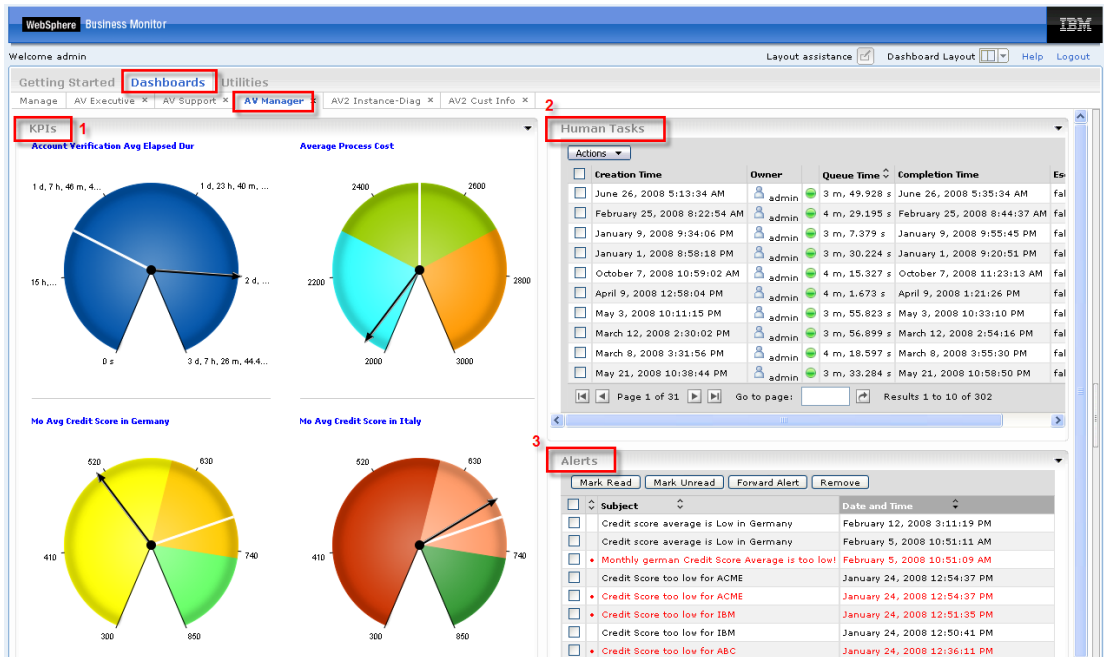
__3.   Have a look at some of the detail in the KPI view on the left half of the layout. This particular view shows four different KPIs displayed as gauges. Later, you will experiment with different ways to graphically display KPIs.

__a. The KPIs are:

__i.   Account Verification Avg Elapsed Dur (duration)

__ii.   Average Process Cost

__iii.   Mo (monthly) Avg Credit Score in Germany

__iv.   Mo (monthly) Avg Credit Score in Italy

__b. Focus on the **Average Process Cost** KPI gauge: Hold the cursor exactly on the white line in the gauge to see the target value for this KPI (2500)

__c. Hold the cursor over any pie section of the gauge to view the definition of that range:

__i.   Poor: 2700-3000

__ii.   Good: 2300-2700

__iii.   Great: 2000-2300

__d. Hold the cursor over the black arrow to find the actual value at the present time. What is the current actual value?

Average Process Cost

hold the cursor here
to see the target

hold the cursor here
to see the definition
of this range

hold the cursor here
to see the actual
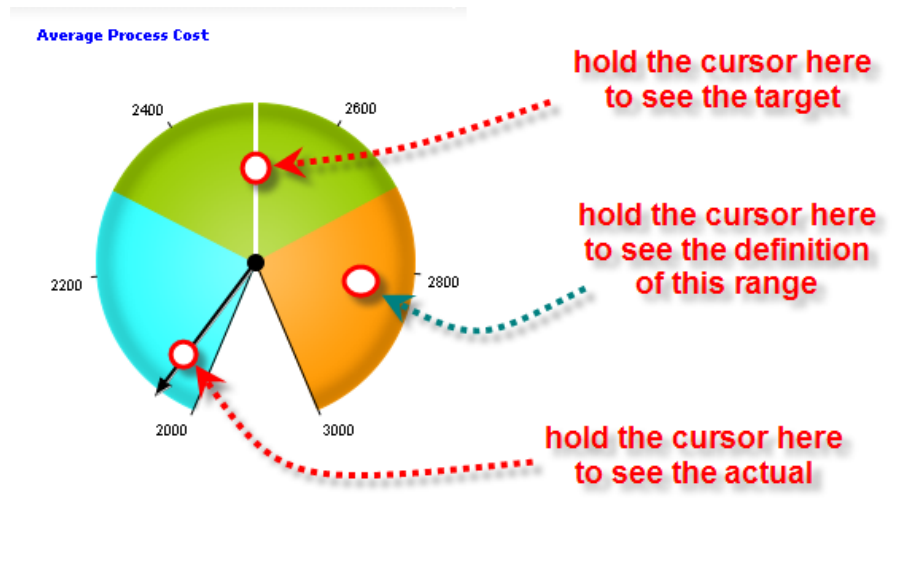
__e. Click once on the gauge on the lower right, "**Mo Avg Credit Score in Italy**". A **Details** panel will emerge. Note the description of the KPI. Click **Go to properties**.

Mo Avg Credit Score in Italy

Alerts

Mark Read   Mark

Details

KPI Name:
Mo Avg Credit Score in Italy

Owner:
uid=admin,o=defaultWIMFileBasedRealm

Description:
this reflects the mo. avg credit score filtered for Italy

Target: Actual value:
700   680

Go to properties...   Close
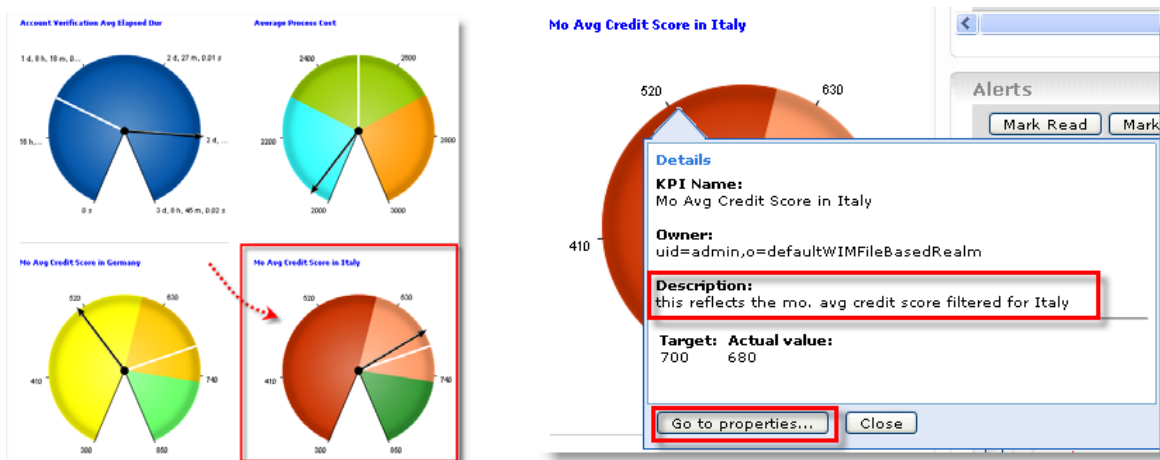
__f. A **Loading KPI Properties** message will appear. The Properties panel will emerge after a short pause, and open on the Name page. Click the tab for **Definition**. This definition page indicates:

__i. The value for this KPI is found by looking at the value of a Metric called Credit Score, then averaging that value across the process instances for the current month.

__ii. The data filter at the bottom of the panel indicates that the results from the above step are then filtered, and only the results from requests where the originating country (a Metric called Customer Country) was Italy.

__g. Click the tab for **Range**. This panel shows the target value and all the currently defined ranges with their limits as well as colors. You be making changes to a similar KPI a little later. Properly authorized users can change all these values and colors if needed.

__h. At the bottom of the KPI properties panel, click **Cancel**.
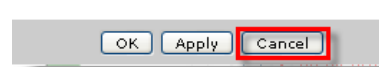
__4. To the right of the KPI view in this dashboard, you will notice a **Human Task** view. It is currently configured to show all human tasks running in the system. In later labs you will look at ways to filter and change the displayed information. Notice the **Owner** column, **Creation** and **Completion** time columns. The green dots in the status column indicate the task is complete.

__5. Click the **Actions** bar at the upper left of the Human Task view. This window allows properly authorized users to take action on a running Human Task or groups of tasks when appropriate. This view is limited to the display of information for Human Tasks. Examine the available options, then click the **Actions** bar again to remove it.
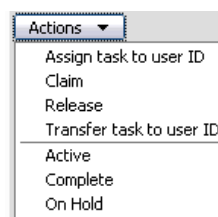
__6. The **Alerts** view is the last one configured on this dashboard. Examine the Alerts view on the lower right. This view demonstrates one of the types of Alerts that WebSphere Business Monitor can send. When a KPI or other measure meets or fails a certain criteria, a Dashboard Alert can be sent (what you see here), or emails, pager or cell messages can be sent.

__a. New items appear in red. They remain red until they are opened, then they turn black, indicated they have been read. Dashboard alerts can be forwarded to other users who for some reason may not be configured to receive the specific alert. At any point they can be removed.
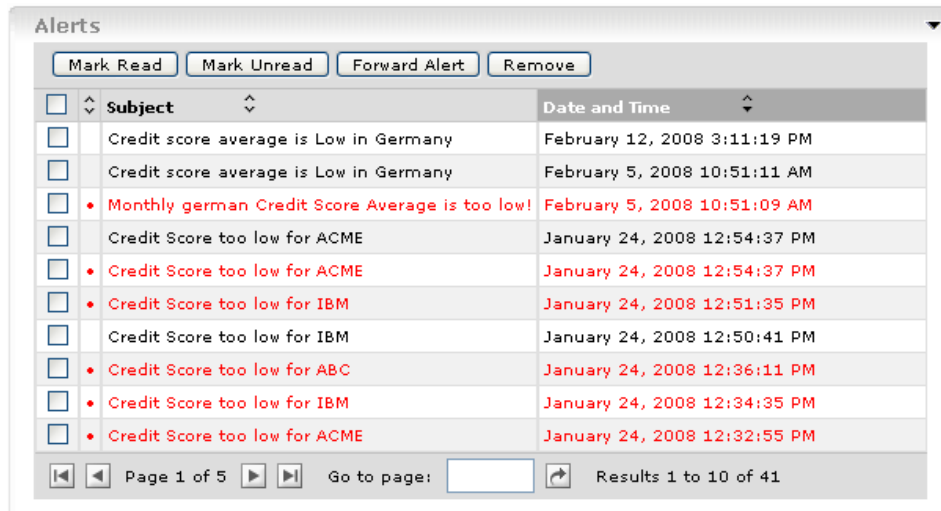
__7. Click a cell with a red dot in it, this indicates an unread alert. A details panel will open for that alert containing the **Subject**, the name of the **Business Situation** event that caused the alert, and any **Comments**.

__a. In the example shown here, the comments have been configured to show dynamic data from the specific offending process, indicating that the actual value at the time of the alert was a Credit Score of **500**.

__b. Click **Close** on the properties panel. Once you have closed the panel, notice the color of the alert has changed to black.

__c. You can place a check next to the item, mark it as unread or read, and the colors will change accordingly. Alerts can be removed or forwarded using the same technique.

__d. In later labs, you will see how to subscribe or unsubscribe to different alerts configured in the system.
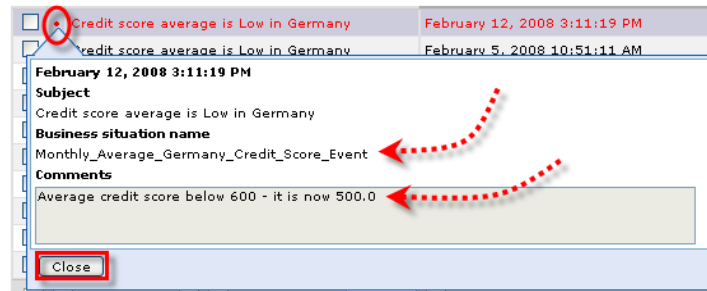
__8.   The **AV Manager** dashboard you have been using is configured with three views:

__a. **KPIs**

__b. **Human Tasks**

__c. **Alerts**

These three views have very specific functions that may be useful to a particular role or type of user. Other roles may need to see different combinations of data that are pertinent to their responsibilities.

__9.   Configuring dashboards is very flexible, you can have multiple copies of the same view on a page configured differently, or you can use completely different views on a single page. You will explore the layout and configuration of these dashboards and views in a later lab.

__10.   This next dashboard in our example has been named the **AV Executive** dashboard, assuming that the information might be appropriate for an executive role. Naming the dashboards according to the role of the user is convenient, but it is optional, you may name the dashboards in any fashion you like. Click the tab for the **AV Executive** dashboard page.

IBM **Tech**Works
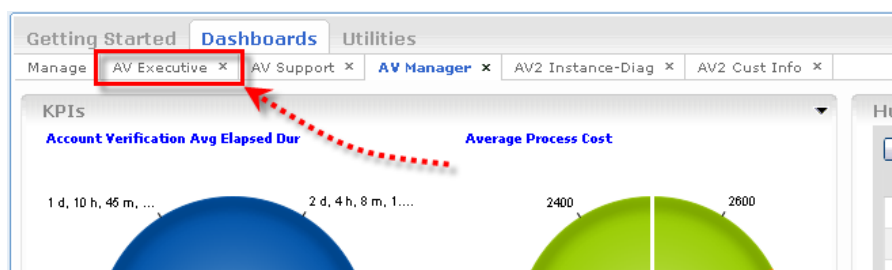
This dashboard contains three views: **1) Dimensions   2) Reports   3) Diagrams**
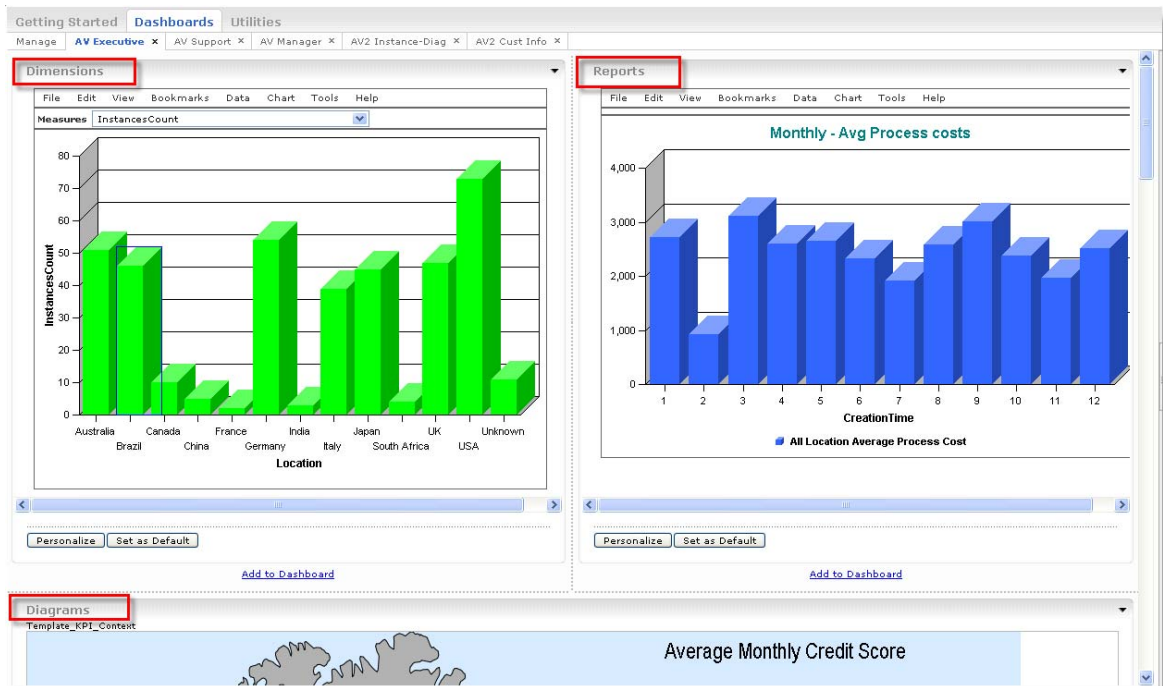
__11.   The Dimensions view allows for examining the data according to the different data "dimensions" that have been specified. (in this example "Location" is the dimension used) Location happens to be a dimension that has two levels: Country and City. The view you see now has been sliced or displayed by Country. The term "drill up" or "drill down" is used to describe viewing the data from a different level, either higher or lower.

__a. Hold the cursor over the column of data showing the number of processes initiated for the **USA**. The actual value appears in the hover help for this column.
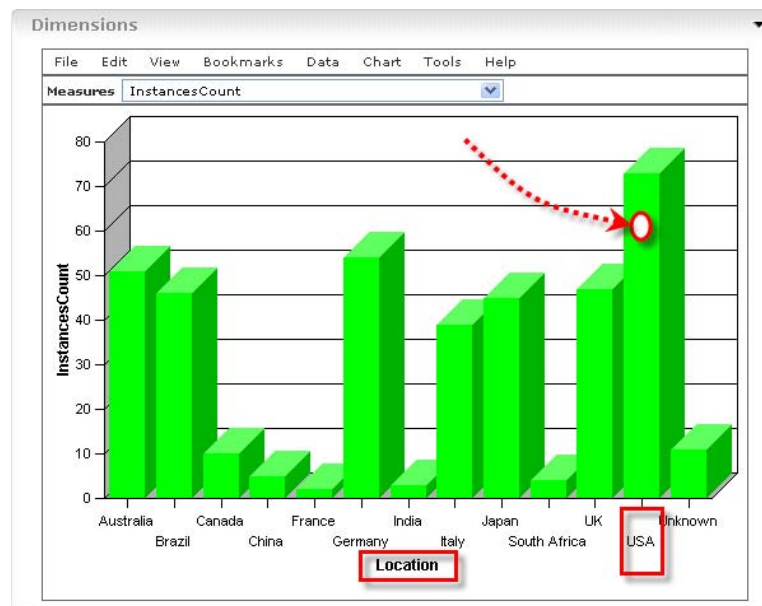
__b. To "drill up" and look at this data for all locations, right click the column and select **Drill Up**. How many processes have been run for all locations?
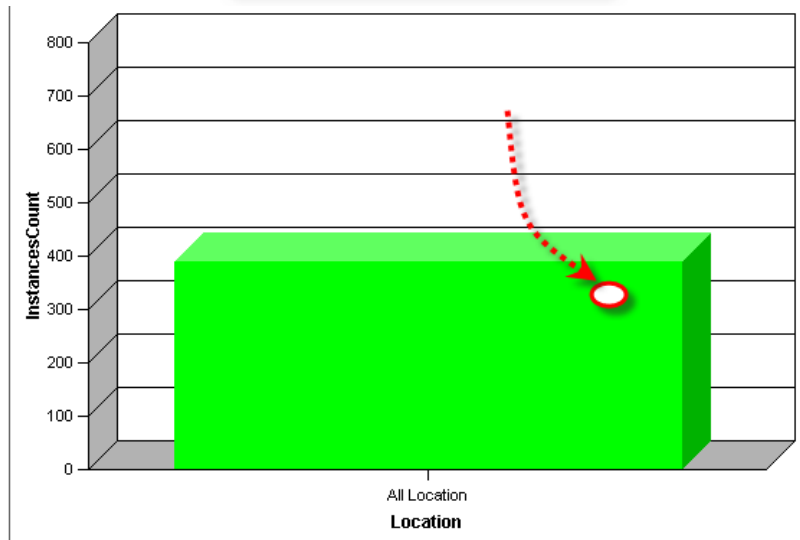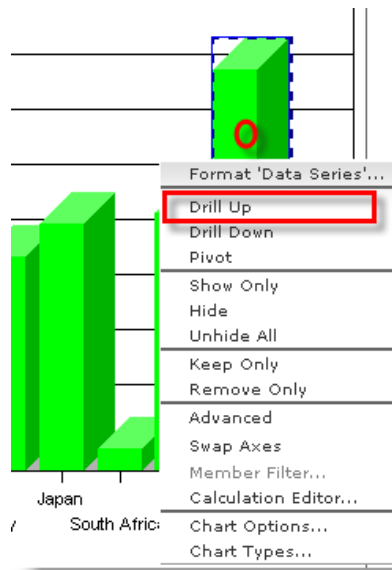
__c. Right click the column of data for All Locations and select **Drill Down**.
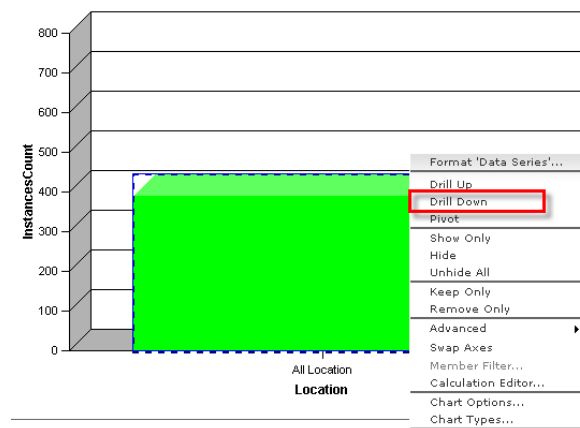
__d. Right click the column of data for the USA as you saw before, and **Drill Down**.
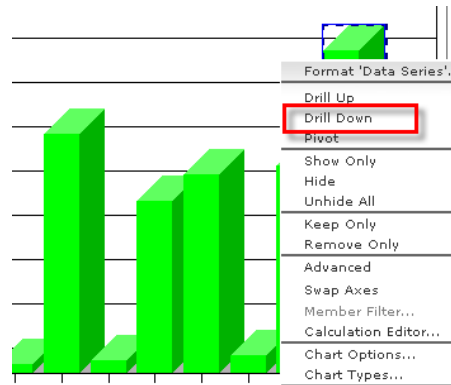
__e. Now you have drilled down into the location dimension, using Instance count as a measure. How many instances (requests) were for customers in New York?
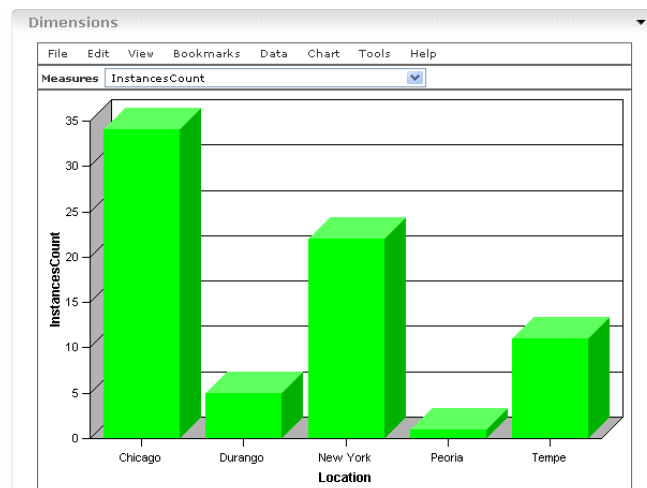
__f. Note the drop down window labeled **Measures**, appearing just above the chart. These measures represent other calculated data that can be used to filter the view displayed.
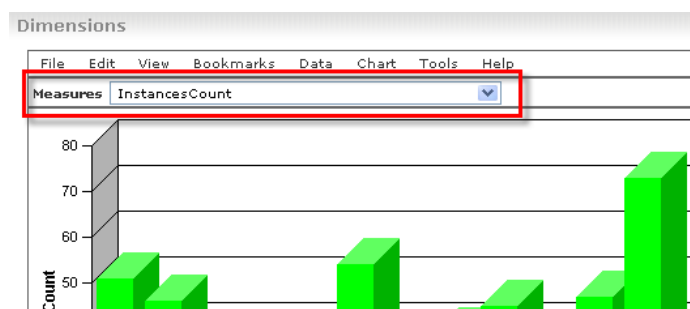
__g. Use the drop down window and select: **Average Credit Score.** The vertical (y) axis now switches from instance count to Average Credit Score.
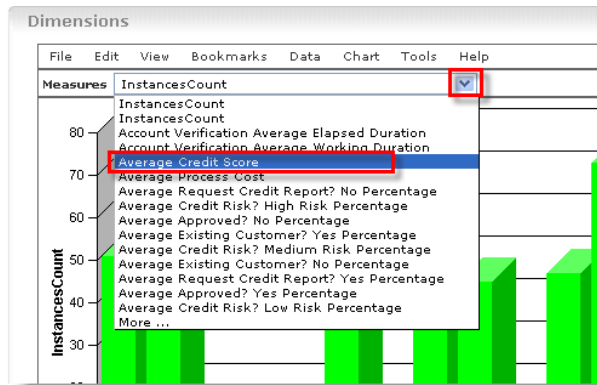
__h. This provides a slice of the data which reveals the **Average Credit Score** by **Country**.
    Again, the hover help displays individual values if the cursor is moved over each column.
    What two countries reflect the highest average credit score for our processes?
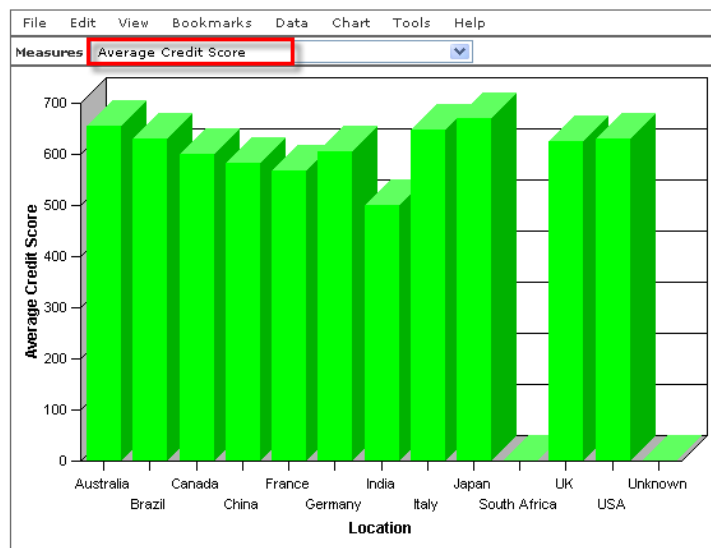
__i. Use the drop down menu again, and select **Average Process Cost**.
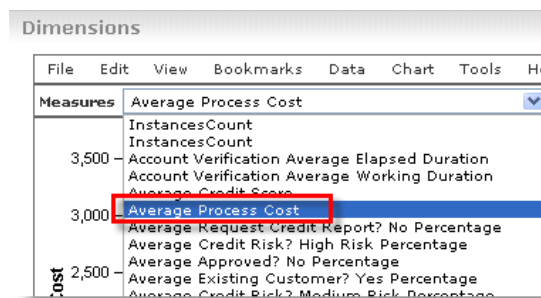
__j. Using this view, what are the three countries whose requests reflect the lowest process
    cost?
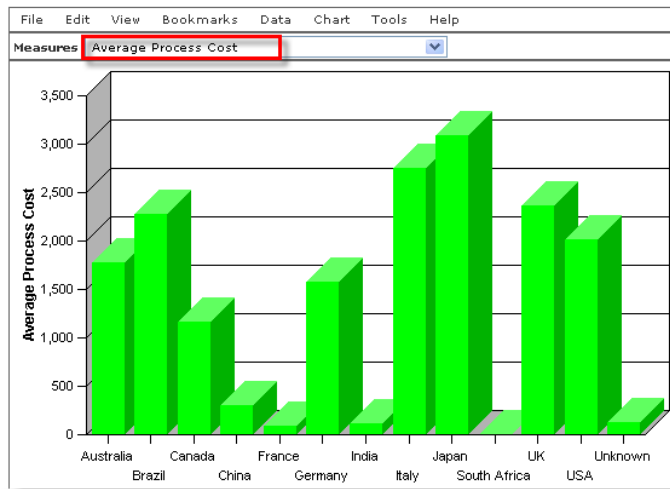
__k. What Cities in the UK reflected the highest process costs? The lowest? Right click the **UK** column and **Drill Down**.
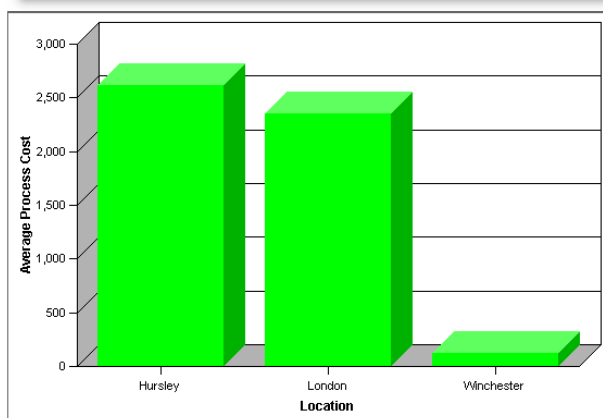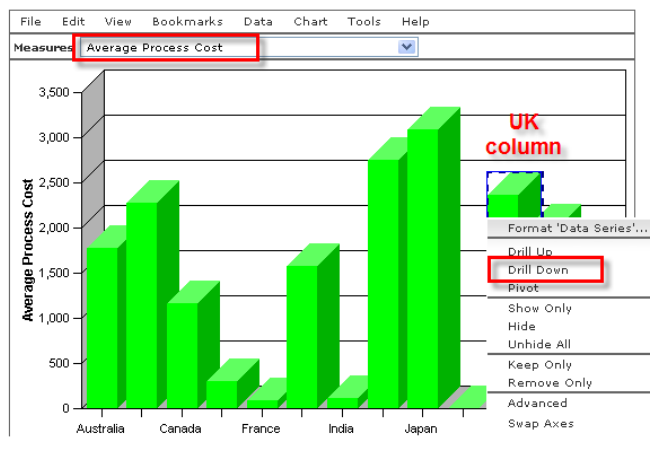
__l. While the Dimensions view you have been working with can be configured with almost any measurement on any axis, the Reports view shows different measurements over time.

__12. Examine the Reports view on the upper right. Currently it is configured to show the Avg Process Costs for each month of 2008. For this example, data has been pre-loaded for all months of the year. The Creation Time label indicates that we are using the date when the process was created. (it could also reflect the end time of the process)

__a. What was the month with the lowest Avg Process Costs?

__b. Right click any data column and select **Drill Up**. Select **CreationTime** and click **OK**. What was the **Average Process Cost** for all of 2008, for all Locations?
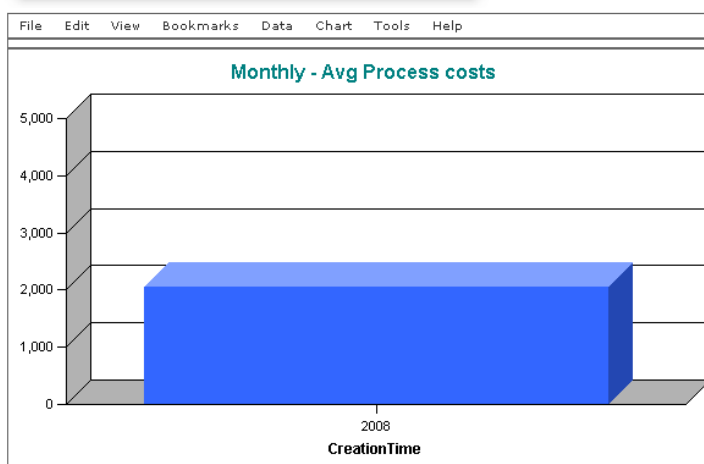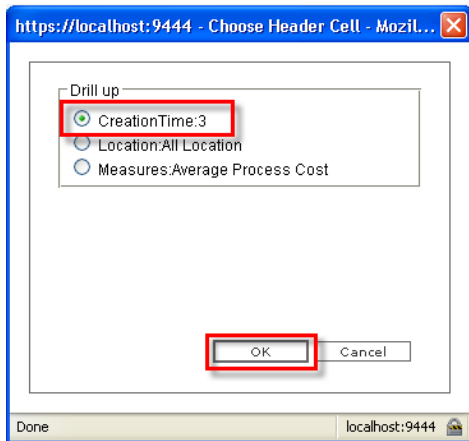
__c. You will see several other configuration options for the report view in later labs.

__13.   One of the powerful capabilities of WebSphere Business Monitor is the ability to use current or historical data it has captured, including calculated measurements and display that data on various diagrams, or use the data to dynamically change parts of the diagram to reflect business performance. The Diagram view you see at the bottom of your display is an example.

__a. Scroll down and examine the world map of Average Monthly Credit Score. This view takes the credit scores of any companies requesting credit through the process, and updates the color of the countries on the map based on the range of their score.
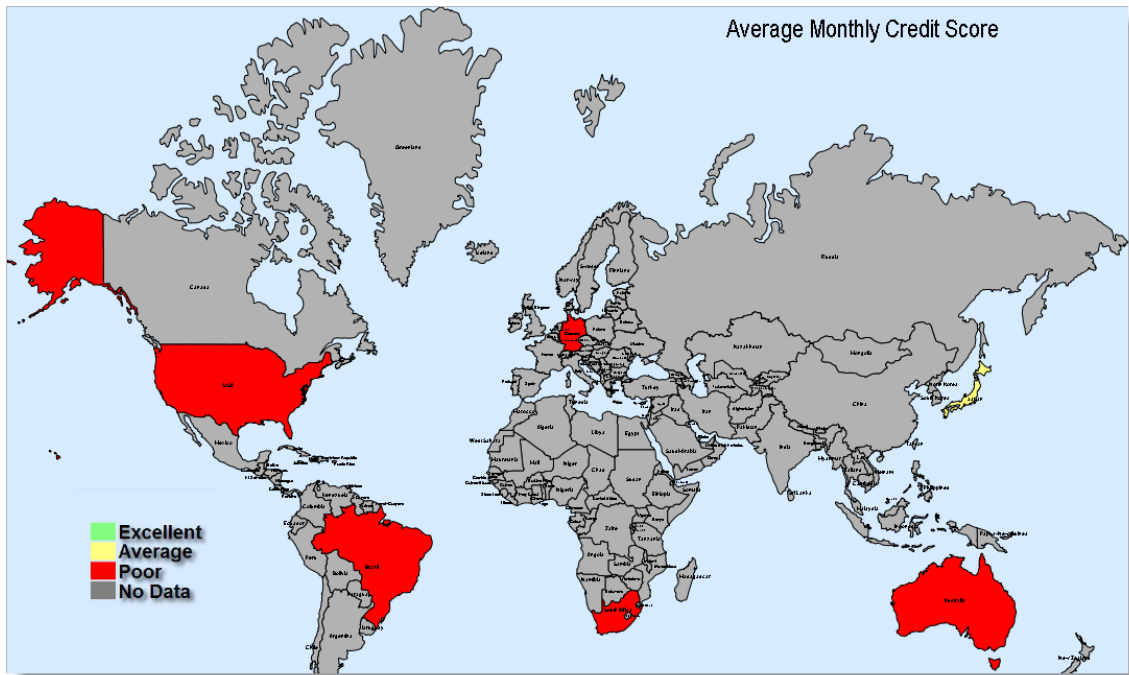
__b. This map is just an example. Any properly formatted SVG (Scalar Vector Graphics) diagram can be used, and based on any of the measurements available in the model, the shapes in the diagram can be colored, actual data can appear directly on the diagram, shapes can appear and disappear based on measurements. Links can also be created so that a shape on the diagram could route the user to another page.

__c. Another example of using diagrams with WebSphere Business Monitor is to display a diagram of the running process, and update this diagram dynamically with process data. You will see this view in a later lab.

__14.   You have now explored a second dashboard (**AV Executive**) which included three views;

__a. Dimensions View

__b. Report View

__c. Diagram view.

__15.   The third dashboard you will examine is named differently, Instead of taking its name from the
        Role of the intended user like Executive or Manager; it has been named according to what
        content it provides.   The **AV2 Instance-Diag** dashboard uses two views:

   __a. Instance view

   __b. Diagram view

   ___xlii.   This is just an example to show another potential combination of views.

__16.   Click the tab to select the **AV2 Instance-Diag** dashboard.

Getting Started **Dashboards** Utilities

Manage | **AV Executive** ✕ | AV Support ✕ | AV Manager ✕ | AV2 Instance-Diag ✕ | AV2 Cust Info ✕
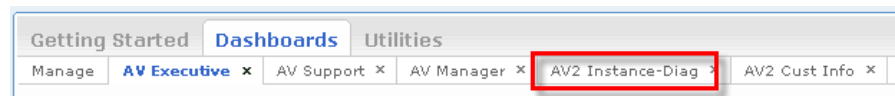
__17.   Notice the upper and lower sections of the layout with the Instances view on top, and the
        diagram view below. Your display may show more or less of the lower view, depending on the
        resolution of your system and the scrolling on the lower panel.
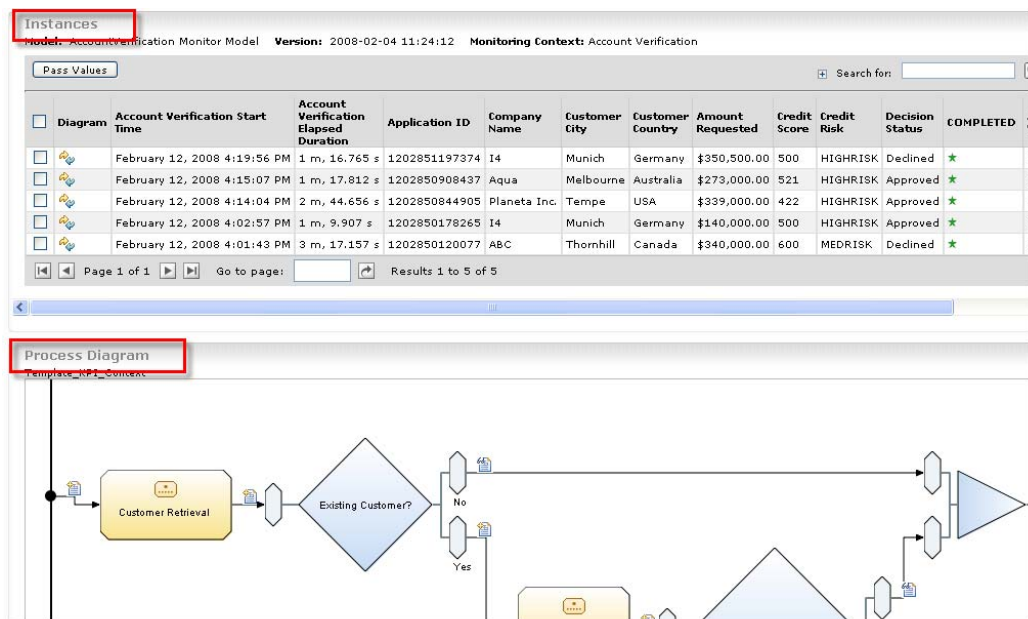
Instances

Model: AccountVerification Monitor Model   Version: 2008-02-04 11:24:12   Monitoring Context: Account Verification
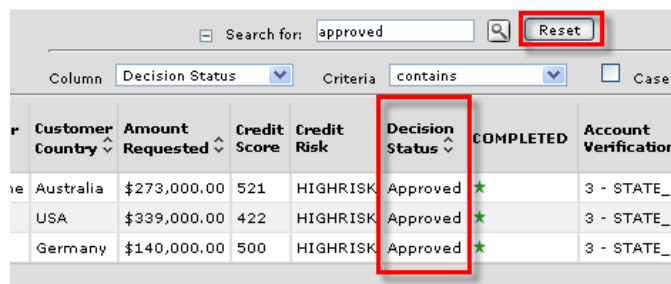
Pass Values

	Diagram	Account Verification Start Time	Account Verification Elapsed Duration	Application ID	Company Name	Customer City	Customer Country	Amount Requested	Credit Score	Credit Risk	Decision Status	COMPLETED
☐		February 12, 2008 4:19:56 PM	1 m, 16.765 s	1202851197374	I4	Munich	Germany	$350,500.00	500	HIGHRISK	Declined	✱
☐		February 12, 2008 4:15:07 PM	1 m, 17.812 s	1202850908437	Aqua	Melbourne	Australia	$273,000.00	521	HIGHRISK	Approved	✱
☐		February 12, 2008 4:14:04 PM	2 m, 44.656 s	1202850844905	Planeta Inc.	Tempe	USA	$339,000.00	422	HIGHRISK	Approved	✱
☐		February 12, 2008 4:02:57 PM	1 m, 9.907 s	1202850178265	I4	Munich	Germany	$140,000.00	500	HIGHRISK	Approved	✱
☐		February 12, 2008 4:01:43 PM	3 m, 17.157 s	1202850120077	ABC	Thornhill	Canada	$340,000.00	600	MEDRISK	Declined	✱

|◀ ◀ Page 1 of 1 ▶ ▶| Go to page: [   ] ↱   Results 1 to 5 of 5

Process Diagram

Template_KPI_Content

Customer Retrieval   Existing Customer?   No

                                                Yes

__18. The Instances view is a display of the occurrences/instances of the Account Verification process that have been run in your system. So far, there have been 5 instances of this process version run. They have all been completed, some have been approved, and some declined.

    __a. Examine the column headings on the Instance view. Note the detailed information provided about each instance in the columns provided.

    __b. Note that the COMPLETED column shows a green star symbol for all those instances completed, and a State column indicating the current State of the process.

    __c. In this process, all the steps are automated except for two tasks that require human intervention. The two Human Tasks are **Request More Documentation** and **Final Application Review**. In this example, we have chosen to use two columns dedicated to display the cost of these two tasks. Since they are human tasks, there is some likelihood that there may be higher variation in their costs, as they depend on human input.

    __d. Use the horizontal scroll bar and investigate all the columns. Authorized users can determine which columns appear in the instance view, and how they are ordered.

Instances
Model: AccountVerification Monitor Model Version: 2008-02-04 11:24:12 Monitoring Context: Account Verification

[Pass Values]   Search for: [   ] [🔍] [Reset]

	Diagram	Account Verification Start Time	Account Verification Elapsed Duration	Application ID	Company Name	Customer City	Customer Country	Amount Requested	Credit Score	Credit Risk	Decision Status	COMPLETED	Account Verification State	Cost - Process	Cost - More Docum
☐	🔍	February 12, 2008 4:19:56 PM	1 m, 16.765 s	1202851197374	I4	Munich	Germany	$350,500.00	500	HIGHRISK	Declined	★	3 - STATE_FINISHED	$51.54	$16.69
☐	🔍	February 12, 2008 4:15:07 PM	1 m, 17.812 s	1202850908437	Aqua	Melbourne	Australia	$273,000.00	521	HIGHRISK	Approved	★	3 - STATE_FINISHED	$50.33	$28.00
☐	🔍	February 12, 2008 4:14:04 PM	2 m, 44.656 s	1202850844905	Planeta Inc.	Tempe	USA	$339,000.00	422	HIGHRISK	Approved	★	3 - STATE_FINISHED	$117.39	$24.63
☐	🔍	February 12, 2008 4:02:57 PM	1 m, 9.907 s	1202850178265	I4	Munich	Germany	$140,000.00	500	HIGHRISK	Approved	★	3 - STATE_FINISHED	$51.49	$22.19
☐	🔍	February 12, 2008 4:01:43 PM	3 m, 17.157 s	1202850120077	ABC	Thornhill	Canada	$340,000.00	600	MEDRISK	Declined	★	3 - STATE_FINISHED	$145.24	$0.00

[◄◄] [◄] Page 1 of 1 [►] [►►] Go to page: [   ] [↵] Results 1 to 5 of 5

information specific to each single instance of the process

__19. The **Search for** function within this view is a quick, simple way of narrowing the displayed instances to answer a quick question about a specific process or group of processes. Do a quick search for all requests that were approved:

    __a. Click the '**+**' to expand the **Search for** section.

    __b. Select the column: **Decision Status**

    __c. Type **approved** in the search for box.

    __d. Click the magnifying glass icon to commence the search:
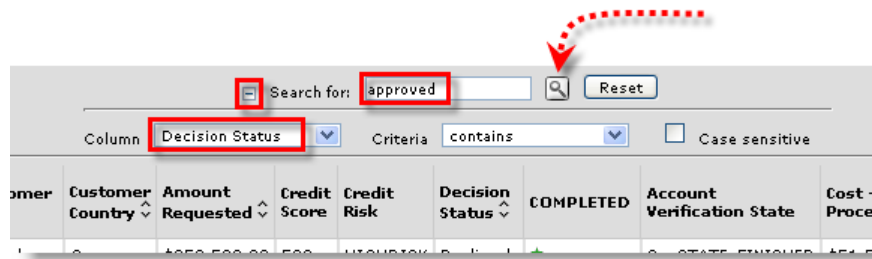
__20. Notice the instances are now limited to those with a Decision Status is Approved. Click the **Reset** button to remove the filter and display all the instances again.

__21. Notice that some column headings display up/down arrowheads next to the heading. This indicates the column can be sorted by clicking on these arrows. Clicking the arrows multiple times will alternate between ascending and descending sorts.
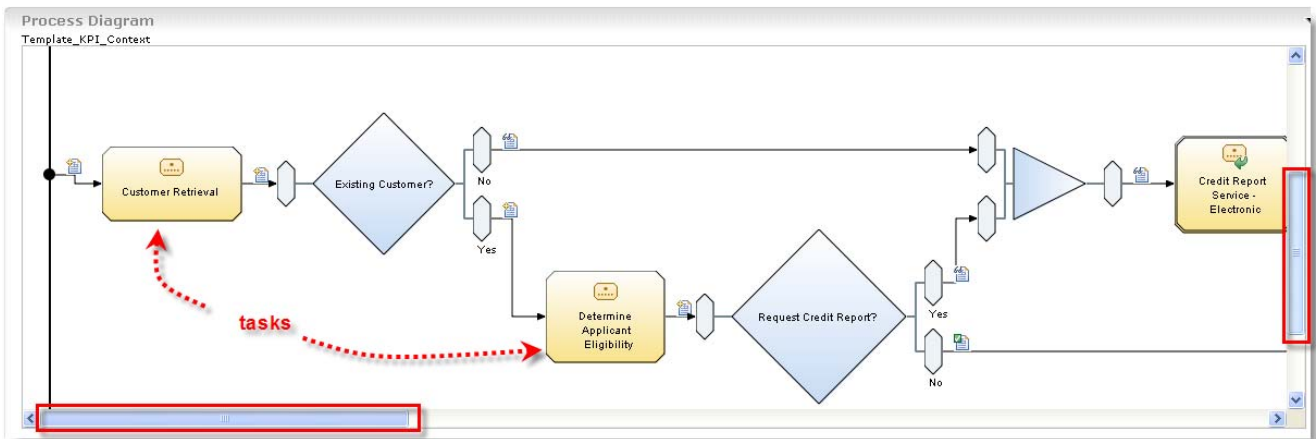
    __a. Click the arrows to sort by **Company Name**

    __b. Click to Sort by **Amount Requested**

heading turns gray when a column is used for sorting

__22. You will see more sophisticated sorting and filtering options available when you personalize the instances view, but for now this provides a brief look at its purpose.

__23. Focus on the **Diagram** view at the bottom of the display. Use the scroll bars to bring the process diagram into view. As you scroll, you may notice the response time can be a bit sluggish. This is due to the fact that you have an entire server system running on your laptop which is constrained by memory and horsepower in comparison to a typical server.

___xliii.   The diagram display was created by WebSphere Business Modeler, and represents the process generating all the data you have been viewing so far. Although WebSphere Business Modeler created this diagram, any SVG (Scalar Vector Graphics) formatted diagram could be used for display here if it helps communicate appropriately with the users.

___xliv.   Notice the tasks and decisions represented in the diagram.

__24. Scroll to the right in your diagram to bring the two Human Tasks - **Request More Documentation** and **Final Application Review** into view. In this mode, the diagram represents all process instances currently in the system. You can also see that at the moment, neither human task has requests waiting to be processed. In a later lab, you will start new processes and see these numbers change as new work begins to queue up. This is an example of WebSphere Business Monitor's capability to enhance a diagram by overlaying real process data onto the image being seen.
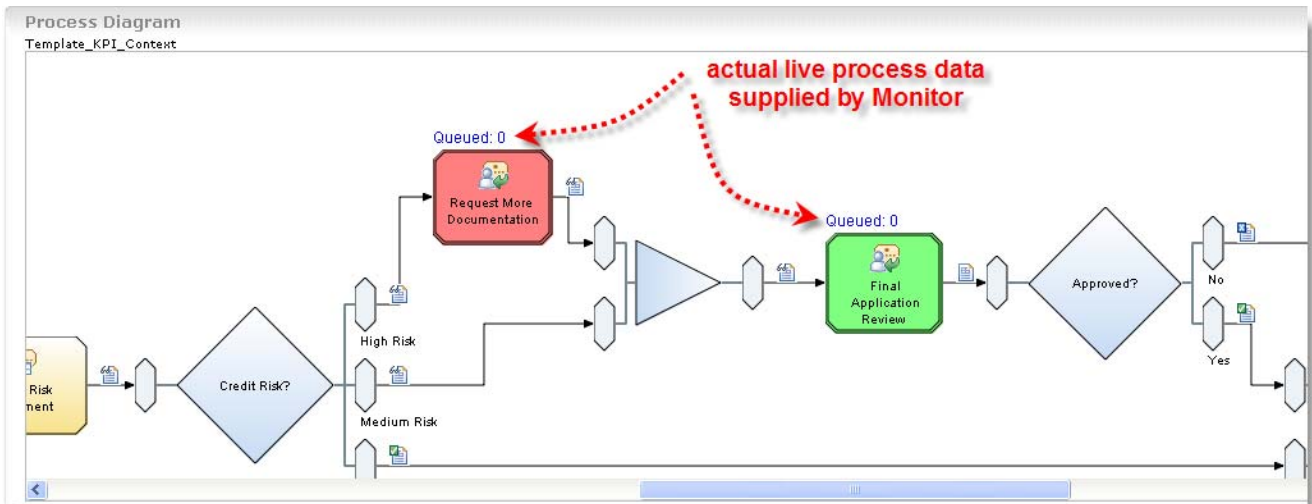
___25. Another example of this occurs when we use the Instance view in combination with the Diagram. In the Instance view at the top of the display, click the **Diagram** icon for one of the processes where the request was **Declined**.

___26. The **Diagram** view is now refreshed and changes the colors of the executed tasks to **blue**. You can confirm by looking at the diagram, the specific path followed in the process. In this instance, since the request was declined, we would expect the Generate Decline task to be blue. In a later lab you will see an example where an alternate color is used to show which tasks are waiting for completion.
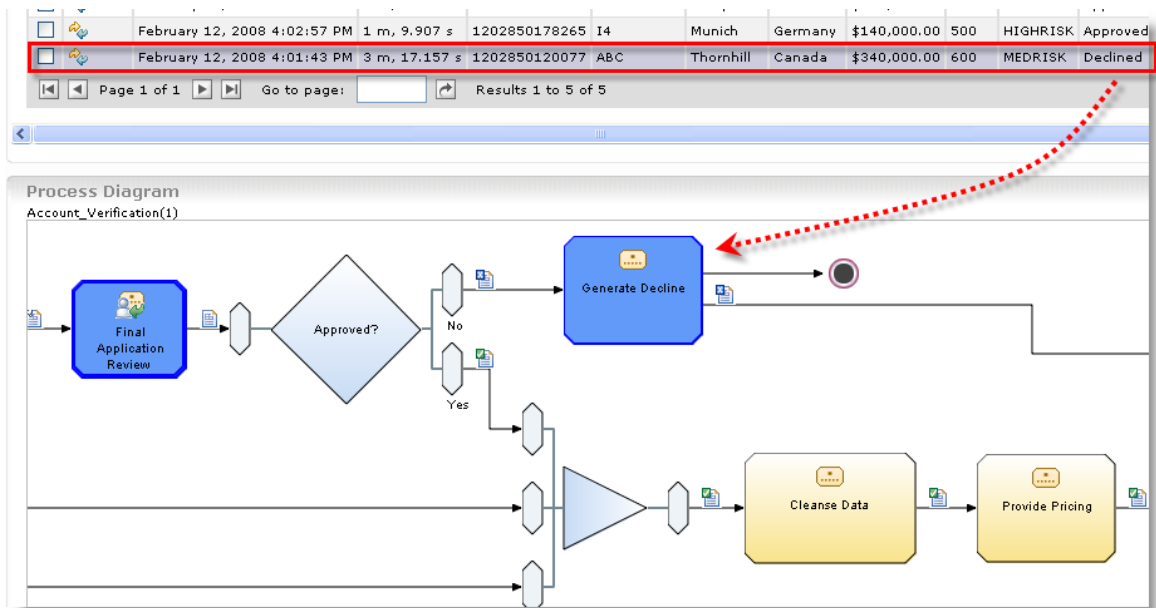
___xlv.   This is an example using a standard process model diagram and very simple behavior. The colors, shapes and behavior of the diagram shapes used are all customizable, so that diagrams that have the best impact with your audience can be used. More detailed diagrams may be appropriate for support or line management, while executive managers may need overview images highlighting a narrow category of exceptions

__27.   In a later lab, you will return to this view and refer to it as you add new process instances and watch the effects.

__28.   Click the **AV2 Cust Info** tab to open that dashboard.
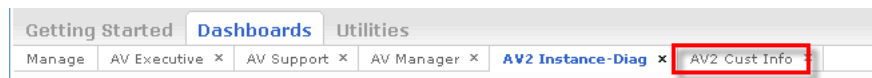
__29.   This dashboard includes three KPI views on the same page. The KPI views all display different KPIs, and use different graphic options to do so. This page is tailored to show information about the customer requests that have been processed. Naming and organization of your pages is completely customizable.

___xlvi.   Three KPIs are included in the **Amount Requested KPI's** view, displayed in Full Gauges

__a. Amt Requested [Apprvd]

__b. Amt Requested [Avg]

__c. Amt Requested [Max]

Two KPI's are included in the **Credit Score KPI's** view, displayed in Half Gauges

__a. Average Credit Score KPI

__b. Avg Credit Score [Mo] Germany

Three additional KPIs are shown in the **Approval KPI's** view, displayed in Bar Graphs

__a. Approval Ratio [Y]

__b. Total Approved [Y]

__c. Total Evaluated [Y]

__2.   Examine the gauges.

__a. Hold your cursor over the various colored sections to get a quick view of the ranges and their names.

__b. Place your cursor over the lines to get actuals or targets. Black Lines/arrows represent current actual values, white lines are targets.

__c. Click the graphical displays to see the **Details** panels for their descriptions.

__d. Use the **Go to Properties** to investigate other properties.
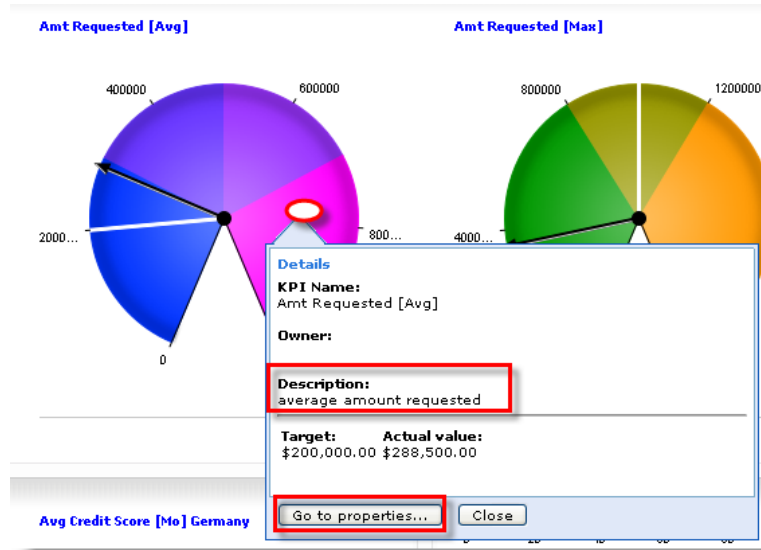
__3.   Click the tab for the **Manage** page. This Manage page is included in the Dashboard section by default, and cannot be deleted. It lists all the available dashboards that have been defined, including those you have been reviewing.

__4.   You will notice that for some of the dashboards listed, there are corresponding tabs at the top of the page. When you click one of the dashboards in the list, it will be displayed, and a new tab will appear for that dashboard title. This way, you can keep the most frequently used dashboards available via the tabs and delete the tabs for those you only use infrequently.

___xlvii.   If you delete the tab (using the 'x'), the dashboard does not get deleted from the list, only the tab is removed.

__5.   Examine the dashboard labeled AV2 Dim I. Click **AV2 Dim I** from the Manage page.
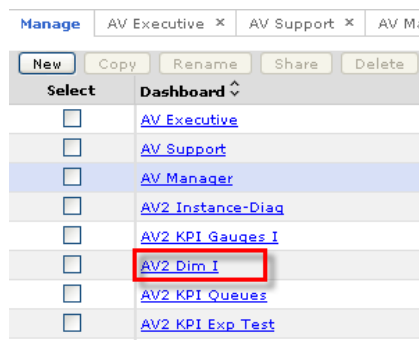
__6.   This dashboard uses a single Dimension view configured initially to see the number of requests (instances) by company, but yet offers several other ways to slice the data.

___xlviii.   This view is configured/personalized with three sets of page dimensions or page filters that can immediately be applied to the page to alter the results that are displayed. The three page dimensions configured currently are:

__a. Measures

__b. Risk

__c. Decision

___xlix.   Initially the chart displays the number of requests (instances) color coded by company for all risk categories and for all decisions (approved and declined), but with a few clicks, many different combinations of analysis are possible.
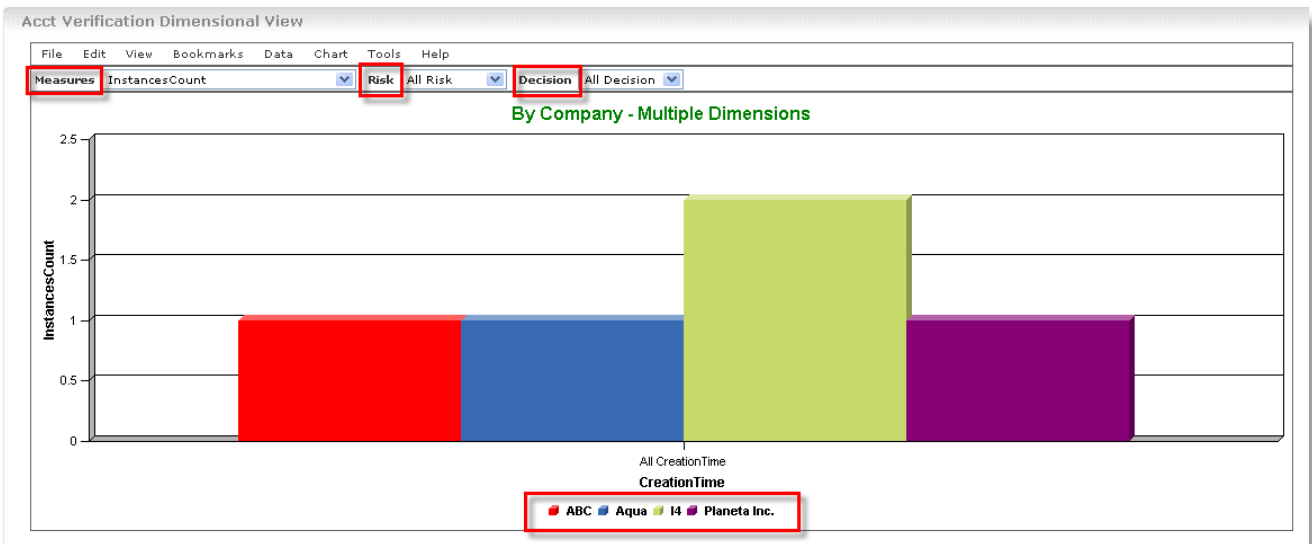
__7.   Change the **Measure** selection from **InstanceCount** to **Credit Score [Avg]**.
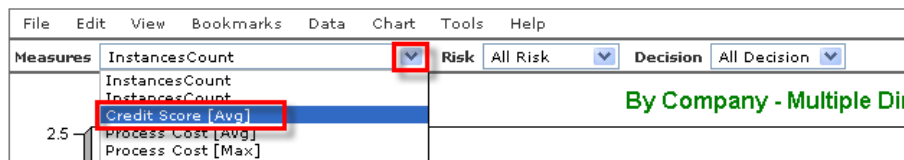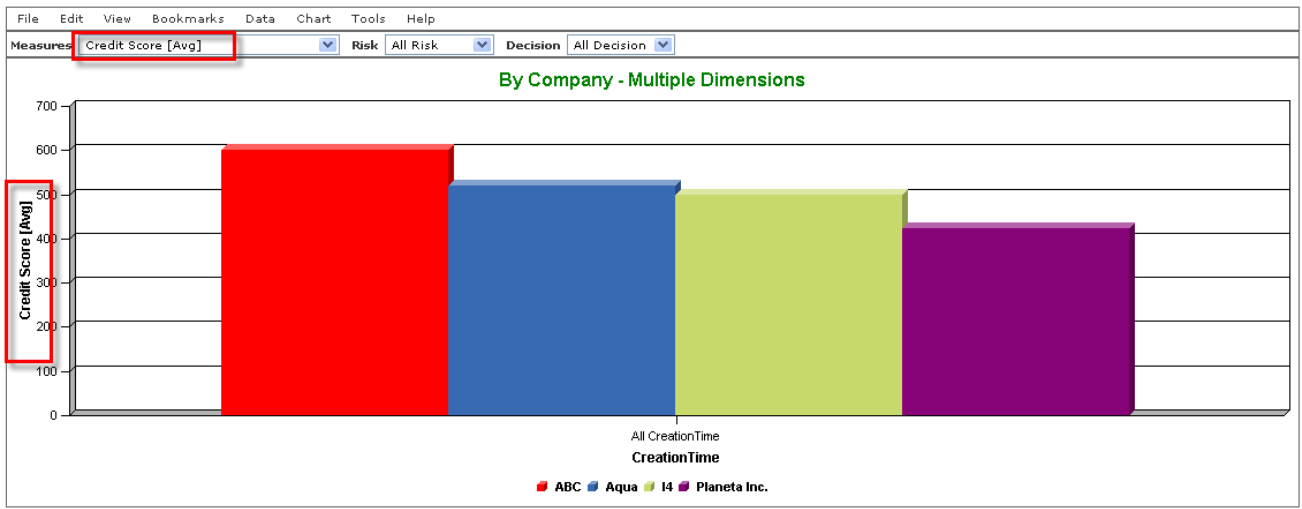
__8.   Change the Measure to examine **Process Cost [Avg]**.
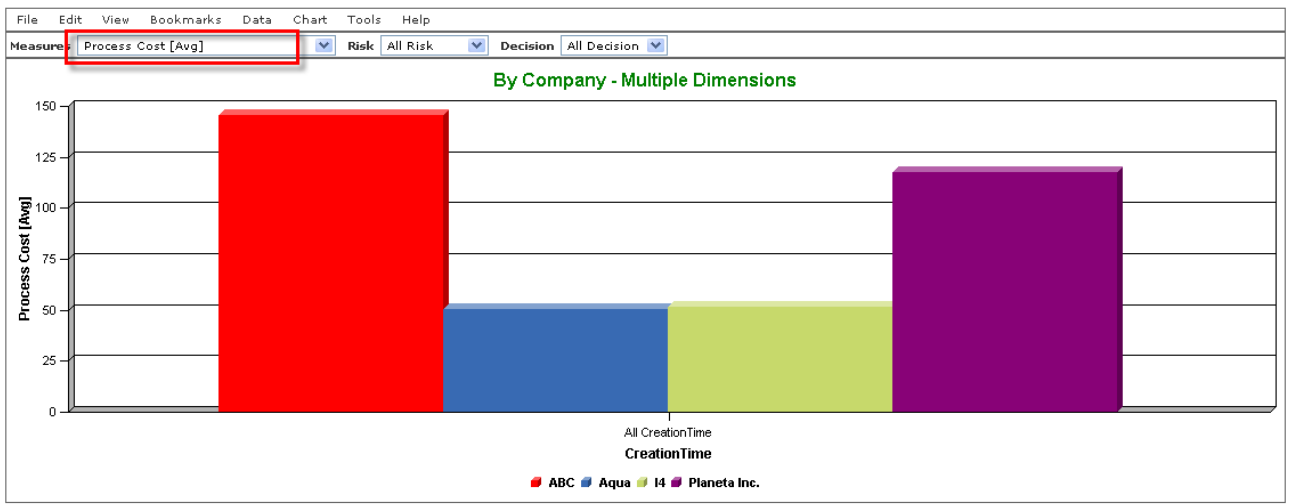
__9.   Limit these results to only those requests with a Decision of **Approved**.
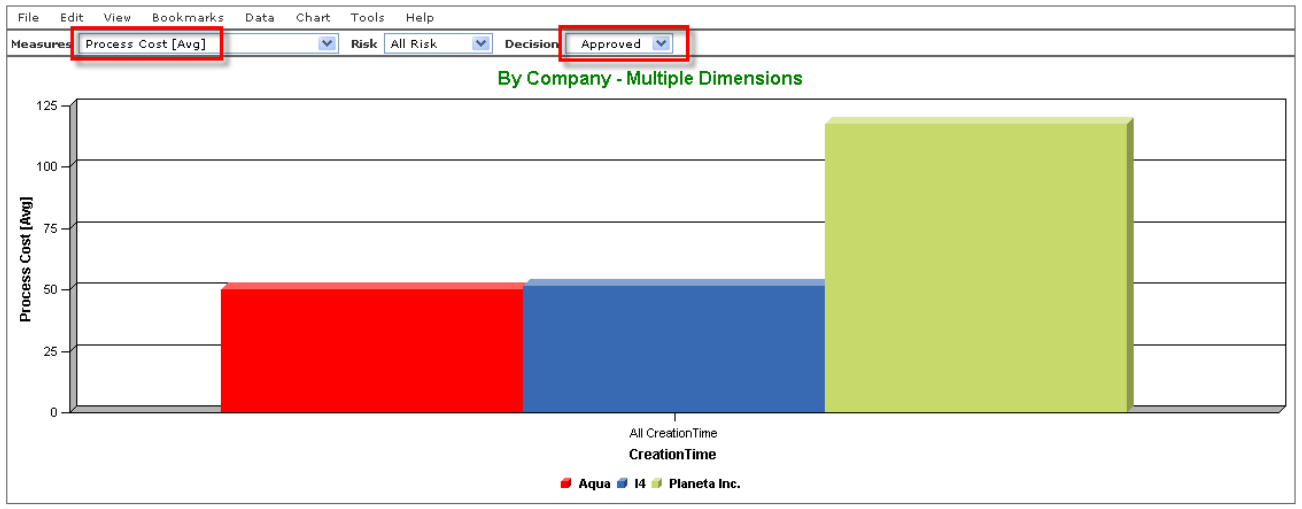
___10. What was the average amount requested by company, for those requests that were declined ? Change the Measure to **Amount Requested [Avg],** set the Decision to **Declined**.

____l. Only two companies currently met this criterion, the ABC Company ($340,000 request) and the I4 Company ($350,500 request).

__11. You can experiment with other selections if you wish. Remember for this example, we only have the 5 existing process instances to gather data from for now. You will add additional requests in a later lab. You may click back to the **AV2 Instance-Diag** page to refresh your memory on the existing instances and the associated data if you wish.

___li.   The point of exploring this view is to emphasize that if useful dimensions and measures selections are provided for the user, the dimensional view becomes a very powerful tool in examining the results and performance of the business process.

__12.   Click the Manage tab again, and this time click the link to the **AV2 KPI Queues** dashboard.
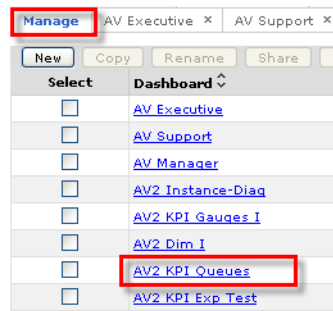
__13.   This page contains a collection of three KPI views, each configured to display a single KPI. The first two gauges display the Queue Depth of the Human Tasks (Request More Doc and Final Application Review) in the process model. Since these tasks are critical to the process, and depend on people to complete them, it may be wise to make sure the number of requests waiting for attention is monitored closely.

___lii.   As you saw from the process diagram view, the text you saw on the diagram Queued: 0 informed us that there were no requests currently waiting for these tasks. These two KPI gauges display exactly that same information, just in a gauge format, rather than on a process diagram. You can see from the gauges that the black arrow – the actual value is now at zero.

___liii.   Also you may notice the orange and red sections, are defined and named slightly differently, and the white lines (targets) are different. The names, range values and colors are purely business decisions and completely customizable. You will do this a bit later.

___liv.   The last gauge measures a yearly total of the number of requests evaluated. The actual value for this KPI – **Total Evaluated [Y]** is now at 5. Confirm this by checking the value of the black arrow with your cursor. This should match the number of instances from the **AV2 Instance-Diag** page, you can confirm if you wish.

___lv.   As you add new process instances later, you will see all these gauges change.
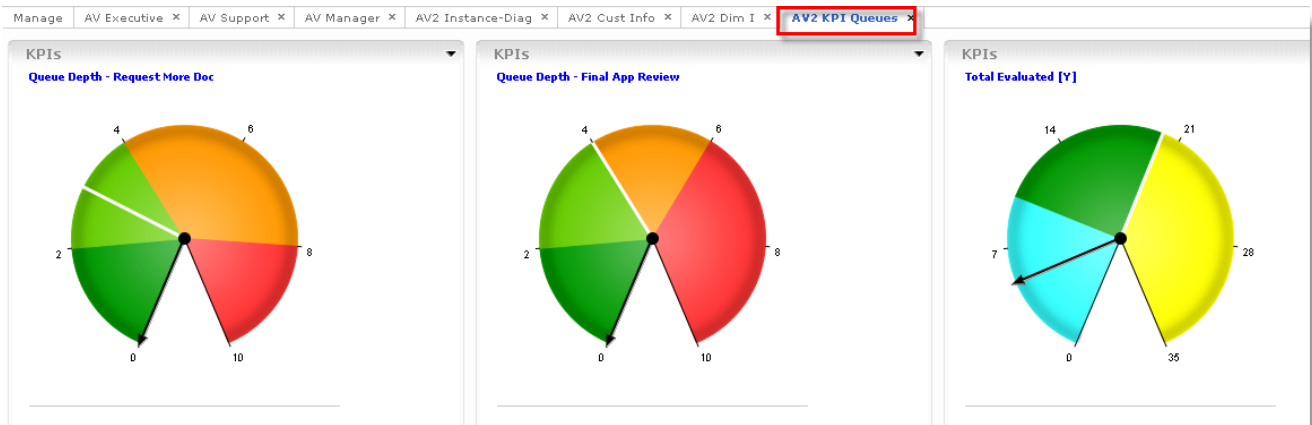
__14. Click the **Manage** tab again, then click the link to select the **AV2 KPI Approval** dashboard page. This page groups several different KPI graphs together to inform the user about the number and percentage of approved requests. You will notice the Daily section contains KPI's all at zero, since there have not yet been any new requests made in the current day. As you add these requests later, this page will be updated reflecting your new totals and the effects on each KPI listed.

___lvi.   As before, explore the ranges, properties, and actuals for the gauges shown if you wish.
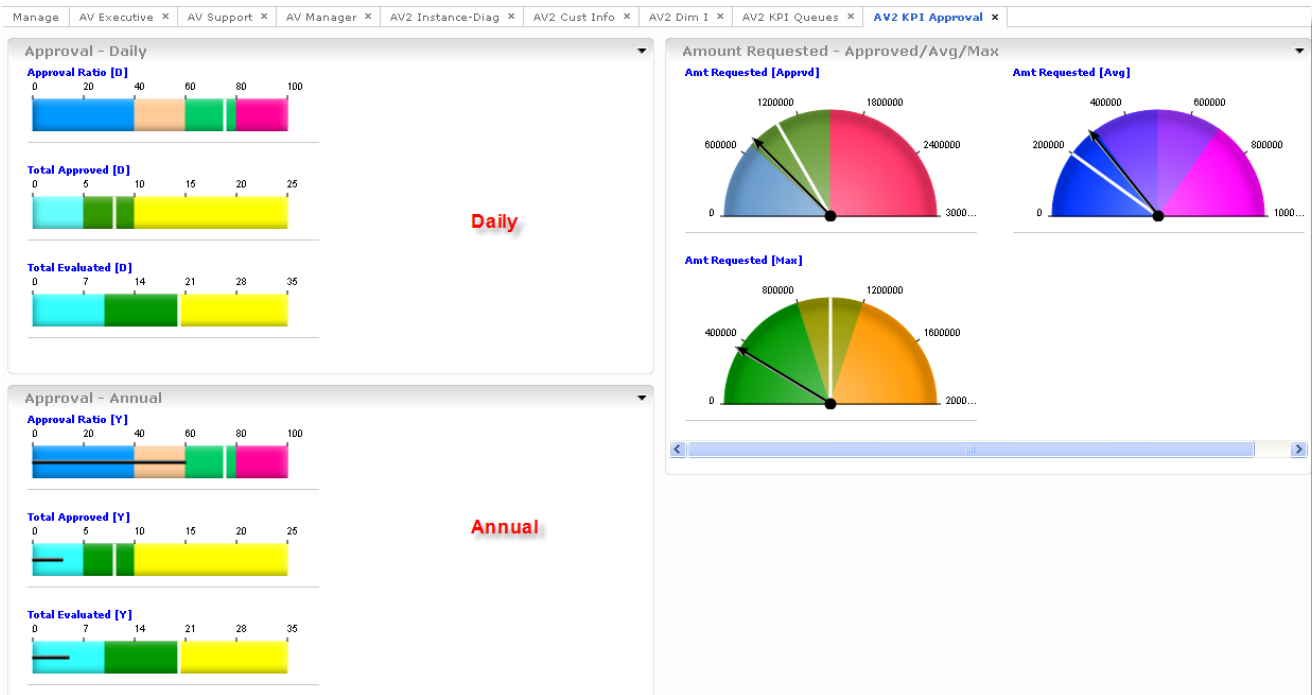
__15.   Click the Manage tab again, then click to select the **AV2 HT Dim** dashboard page. This page has two views; both are Dimensional views of the two Human Tasks in the process. The views are configured to display slightly different information, but both reveal differences between how the owners of the task perform.

___lvii.   The Request More Doc view displays the number of task instances handled by each of the owners who claimed the tasks: bdaniel, mcharles and redwards.

___lviii.   The Final App Review view displays the elapsed duration that each of the owners took to complete the task. Owners of this task included: admin, bdaniel, mcharles and redwards.

___lix.   The All Status is displayed now, but later when you have some tasks complete and others running, you will be able to drill down by status and make further comparisons.
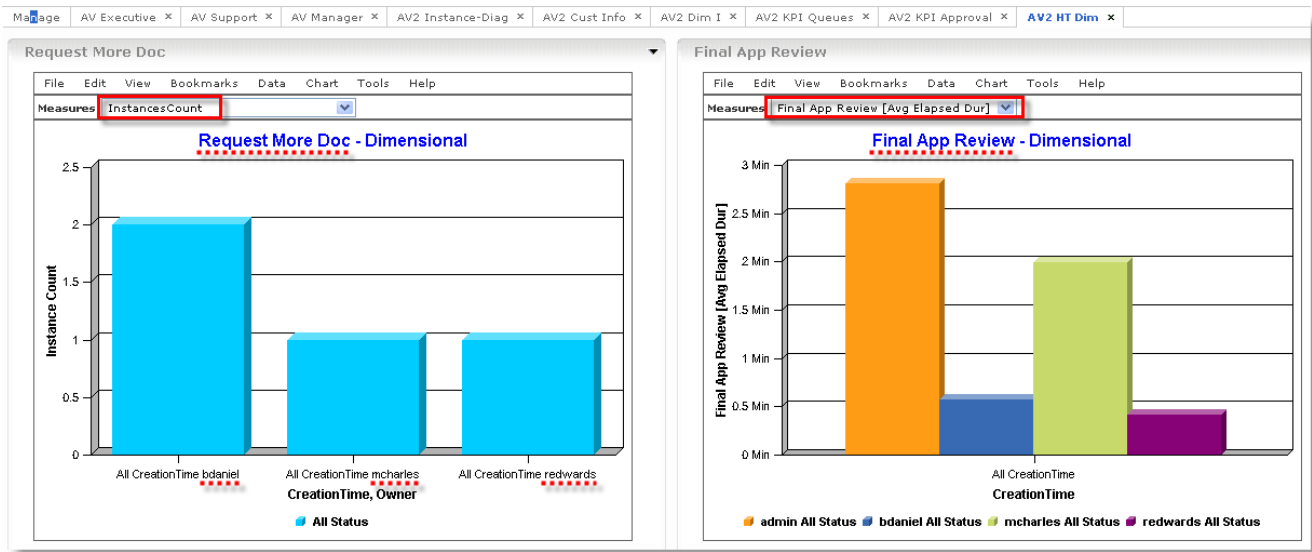
__16.   Click the Manage tab again and click the link for the **AV2 Dim II** dashboard. This view is configured to display a pie chart, which provides a percentage view of any of the filtering choices selected.

___lx.   Choose some different selections for Measures, Decision and Risk, and view the results.
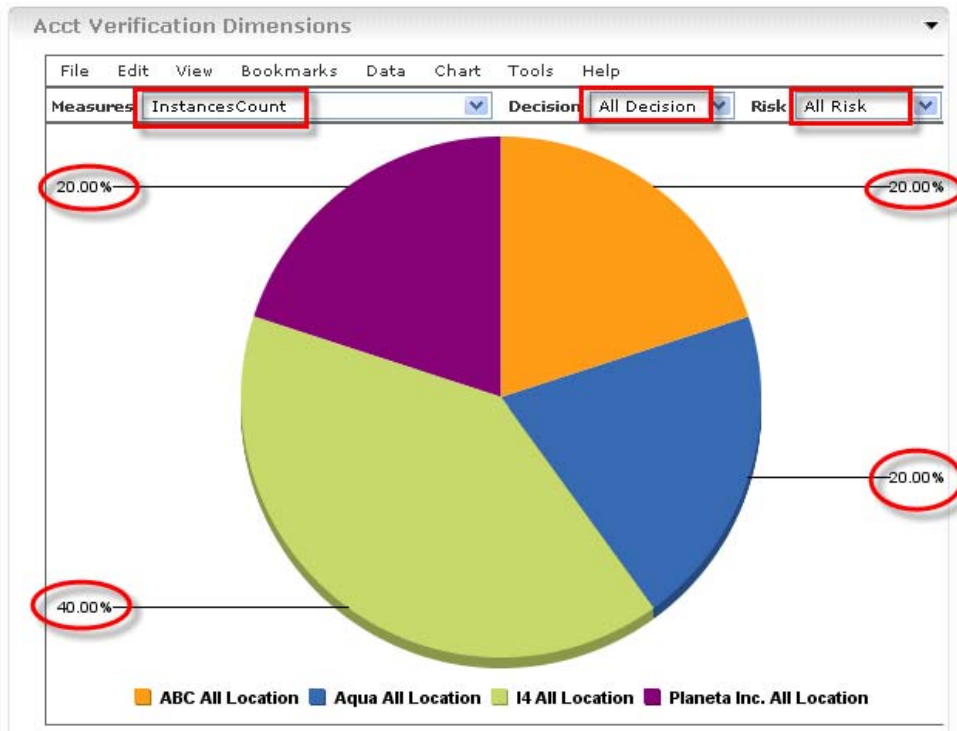
__17.   You have briefly reviewed several examples of some of the dashboard configurations possible using WebSphere Business Monitor. In subsequent labs, you will have the opportunity to:

    __a. Examine some of options available to personalize the dashboards, create new dashboards and change layouts.

    __b. Submit live requests to the system for processing, and see the real-time effects on the dashboards.

    __c. Create new and manage existing KPI's used on dashboard pages.

    __d. Create the preliminary Business Measures that you see available in the existing dashboards using WebSphere Business Modeler. (business participants)

    __e. Edit a Monitor Model including the metrics, measures, KPI's you have seen using the WebSphere Integration Developer – Monitor toolkit. (technical participants)

__18.   Before leaving this section. Delete some of the dashboard tabs from view to simplify navigation. The dashboards will still exist on the list in the Manage tab, and the tabs will reappear when you reference them again later.

    __a. Click the 'x' to remove the **AV2 Dim II** tab from view
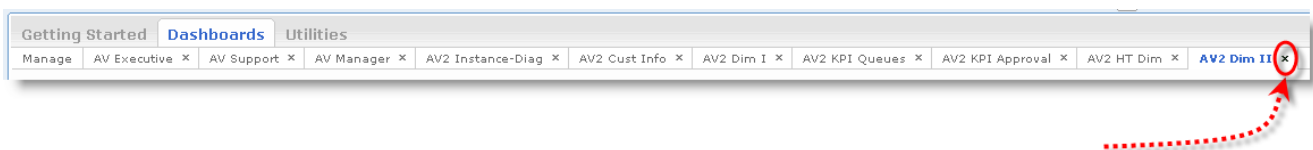
    __b. Click the 'x' to remove the following tabs from view:

        __i.   **AV2 HT Dim**

        __ii.   **AV2 KPI Approval**

        __iii.   **AV2 KPI Queues**

        __iv.   **AV2 Dim I**

__19.   You have completed this lab.

Lab 15.2 - Create and Personalize Dashboards

In this lab, you will create a new dashboard page and add views to the page. Layout options will be reviewed. The personalization options for some of the views will be explored, and you will select the information and formatting of the data for display.

Create a new dashboard

As you have seen, dashboards are collections of views or view items. The views are placed on the dashboard using a layout that is chosen, based on the appearance and screen arrangement you feel is most effective for the target audience for the dashboard. Dashboards can be personal or they can be shared with others in the organization.

To review, Views can contain different types of information as you have seen:

___12.   Key Performance Indicators – this view accommodates one or more KPI's displayed using different graphical objects; gauges, bar graphs.

___13.   Dimensions – a view that presents charts containing different categories that can be used to slice the data – by Company, by Country

___14.   Instances – a way to look at individual occurrences of a process and the related detailed data

___15.   Reports – similar to a dimension view, but less configurable and includes the notion of looking at the results over time

___16.   Alerts – certain business situations have been modeled to send alerts when defined criteria is met, this view simply displays these alerts for subscribed users and offers a way to mark them read or unread

___17.   Diagrams – this view can display any SVG (scalar vector graphics)

___18.   Human Tasks – this view, especially tailored to human tasks from the process shows the owner, disposition and various times associated with tasks requiring human handling.

Dashboard design, naming, organization and techniques for presenting different types of data can be a topic unto itself. We are not dwelling on all these factors, nor attempting to reflect any specific set of best practices. The goal here is just to demonstrate the capabilities of the product. Some of the examples may or may not be ideal for your environment and user community.

__1.   Create a new dashboard page containing two views, a KPI view and a Dimension view.

__a. Click the **Manage** tab to see the list of existing dashboards.

__b. Click the **New** button.

__c. Type **MyDashboard1** for the Name of the new dashboard. Use the drop-down arrow to select a layout with two views side by side, as shown below, note the other options available.
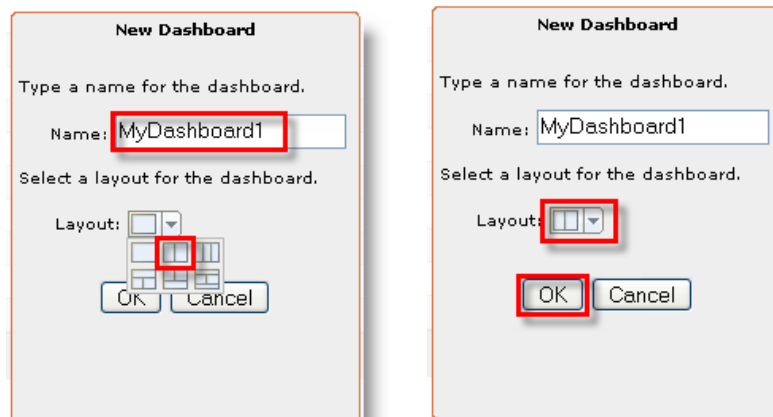
__d. The new page layout is now displayed with a divider shown, awaiting the addition of the two views. A new **MyDashboard1** tab is also created.
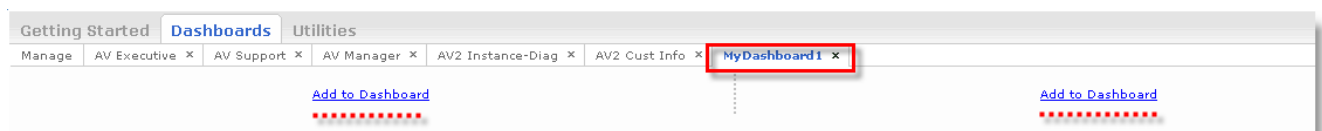
__e. Behind the scenes, a new **MyDashboard1** entry has been added to the Manage tab.

__f. On the **MyDashboard1** tab, there are two ways to add new view items to the page. First, you will use the palette to drag a new view onto the dashboard page.

__2.   Click the palette control on the far right of the display (vertical bar). This displays the palette of view items which can be dragged to the page. Clicking the vertical bar toggles the palette on and off.
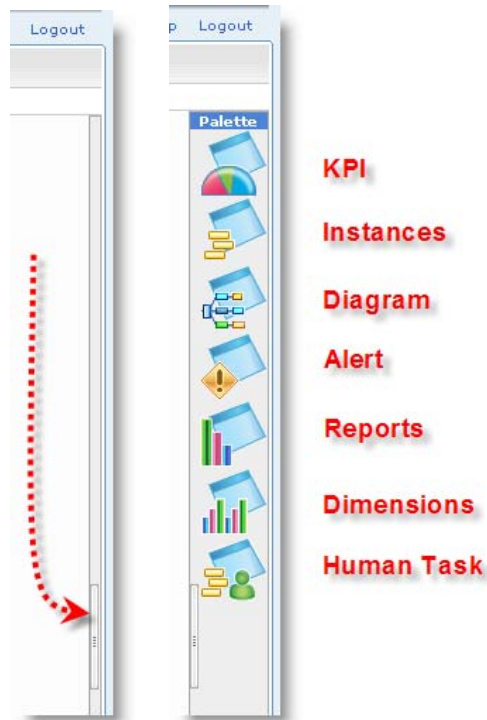
__3.   Drag and drop the KPI icon from the palette to a position *above* the blue Add to Dashboard link shown at the left of your page. When you reach a location where the view item can be dropped, a black horizontal line will appear above the link.
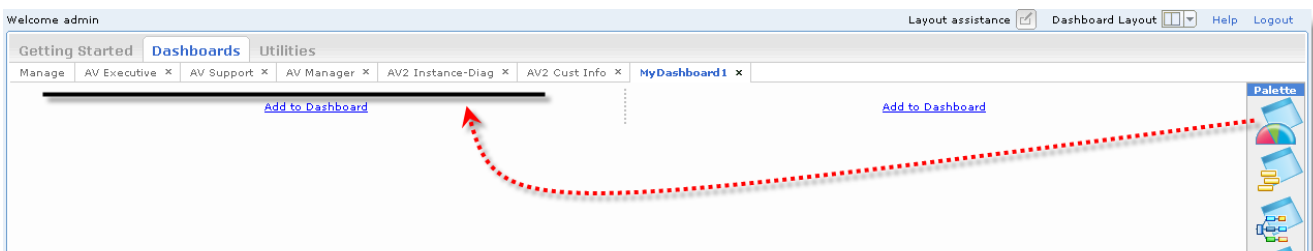
__4.   The new KPI view has been created, but it is not yet personalized. Before personalizing this view, add the second view to the page.
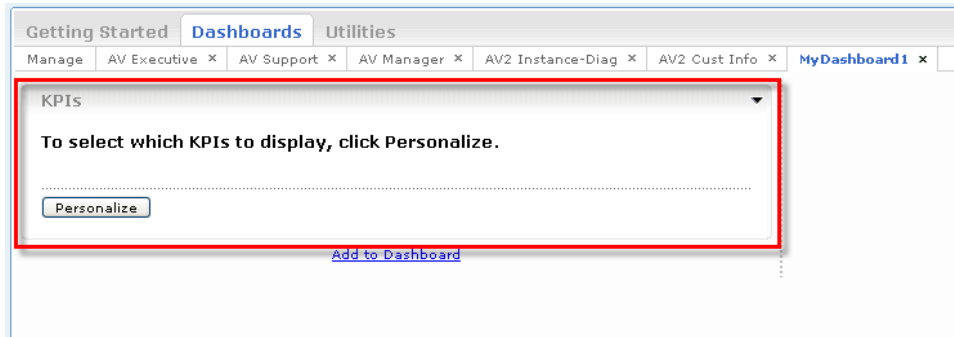
__5.   Click the vertical bar on the view icon palette to switch it off. You will now use the second method for adding a Dimension view to your dashboard page.

__6.   Click the blue Add to Dashboard link on the right side of your dashboard page.
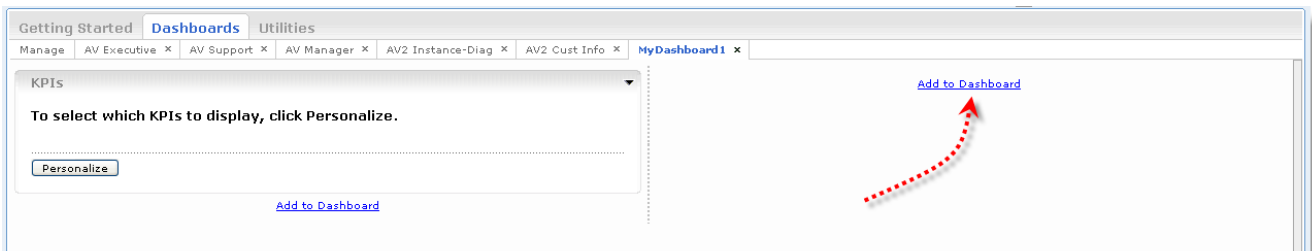
__7.   Use the drop down arrow to select **Dimensions**, click OK.
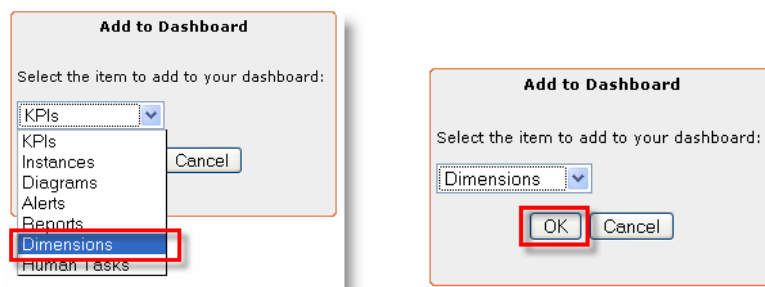
__8.   The new Dimensions view has now been added to the right side of the **MyDashboard1** page.
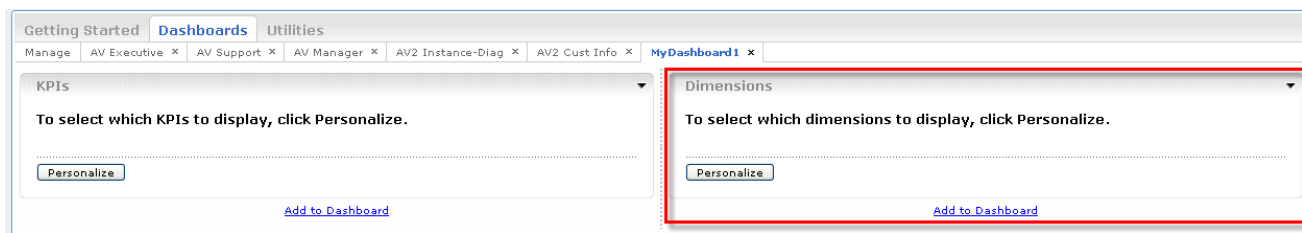
__9.   You have now created the new **MyDashboard1** page, and added to views, a KPI view and a Dimension view to the page.

Personalize the new views on the dashboard

In this section you will specify the model that defines the data you want to display in the view. Essentially, you are telling the view how to find the data it needs. In addition to the data source / model, you will specify some layout options. Depending on the type of view, there may additional personalization-configuration options available to work with.

There may be several models running in your system, representing many different processes or other data sources, in your lab system three monitor models are available. You will use the most recent model.

__1.   Add the individual KPI's to the new KPI view. Based on your conversations with the role or audience for this dashboard, you have determined that four KPI's are needed to best meet the needs.

   □   Account Verification Average Working Duration – the average time spent working on this process

   □   Amt Requestsed [Apprvd] – the total amount of credit extended for all requests

   □   Approval Ratio [Y] – the percentage of requests that get approved

   □   Average Credit Score KPI

__a. At the lower left of the new KPI view you have added, click the **Personalize** button.
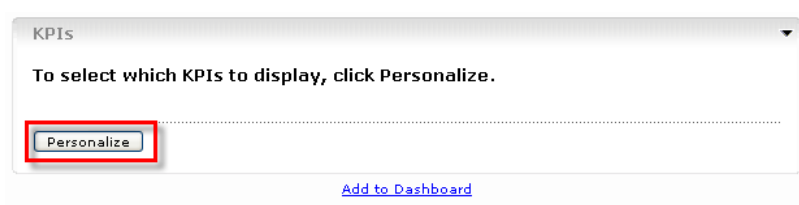
__b. The Select KPIs panel will open. Click the '**+**' to expand the **AccountVerification_Monitor_Model 2008-02-04 11:24:12** model.

> **NOTE:** the position of this model in the list may be different than shown, so make sure you select the correct entry, regardless of where it shows up in the list.

___lxi.   At this point you can scroll and see all of the KPI's that were included in this monitor model. This includes KPI's created in WebSphere Business Modeler, and additional KPI's added by the Integration Specialists at implementation time.

__c. Place check marks by the four KPI's you need for this KPI view. **DO NOT** click OK yet.
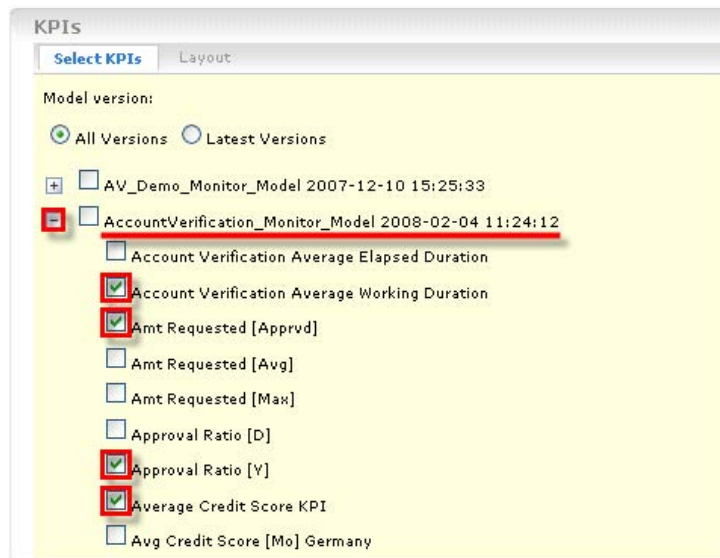
__d. Notice that a Layout option is also available in the KPI view. Click it now and select the following:

> __**i.**   Display Mode: **Full Gauge**
>
> __**ii.**   Layout: **Grid**
>
> __**iii.** Columns: **2**
>
> __**iv.**   Size: **Large**

__e. Now with the individual KPI's selected and the layout specified, click the **OK** button at the lower left. The four KPI's now appear in the view.
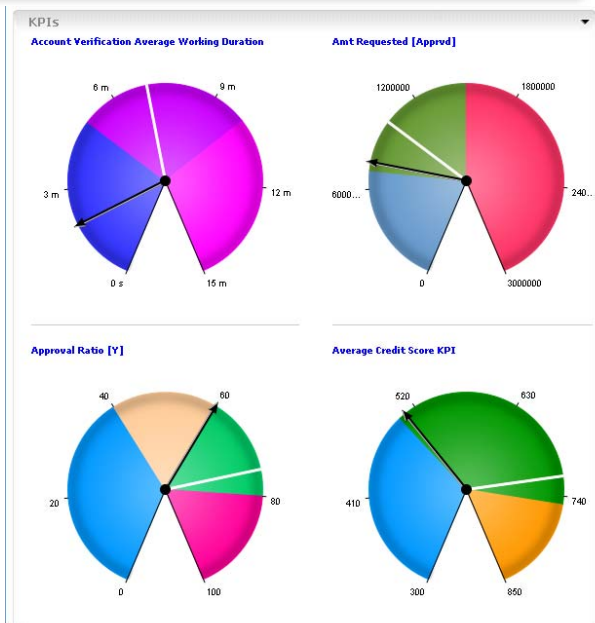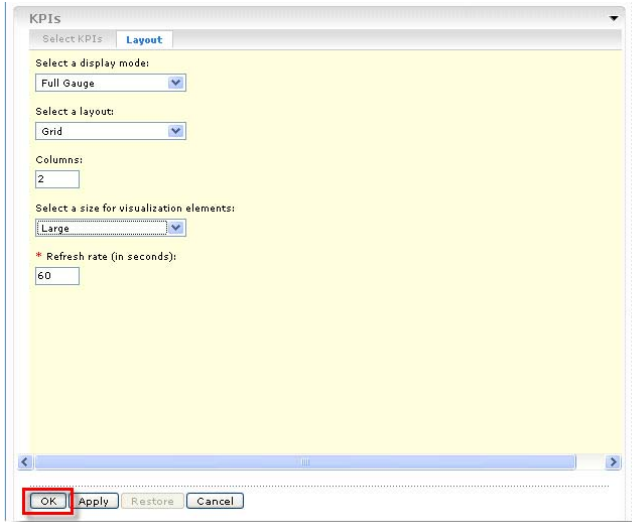
__f. Personalize the label for the view by clicking the label **KPIs** at the upper left. Type a new label **My KPI View 1,** click **Save.**
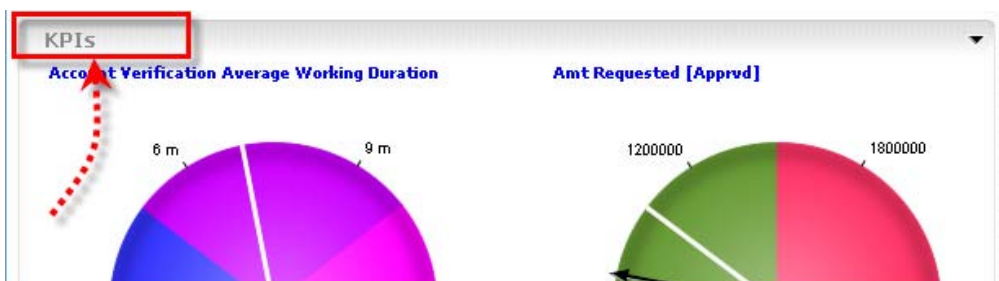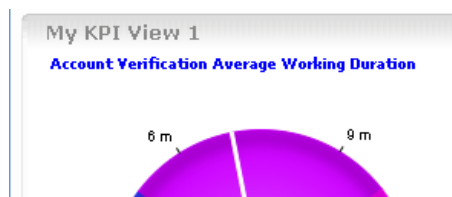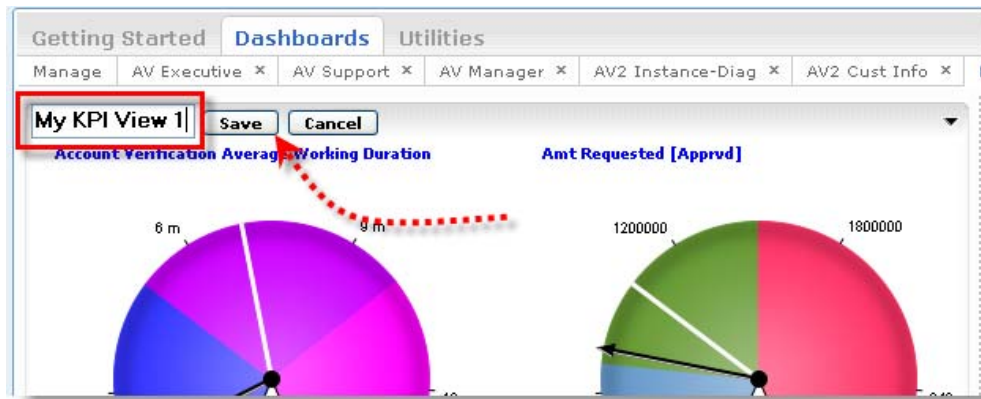
__2.   Personalize the new Dimensions view.

__a. In the Dimensions view, click the **Personalize** button.

___lxii.   You have determined in your conversations with your audience, that a pie chart by company, with the capability to filter by request or instance count, process cost, decision and risk would meet the need.

__b. Use the drop down arrow and select the monitor model to be used for this view, again choose the 2008-02-04 11:24:12 entry for the AccountVerification Monitor Model.
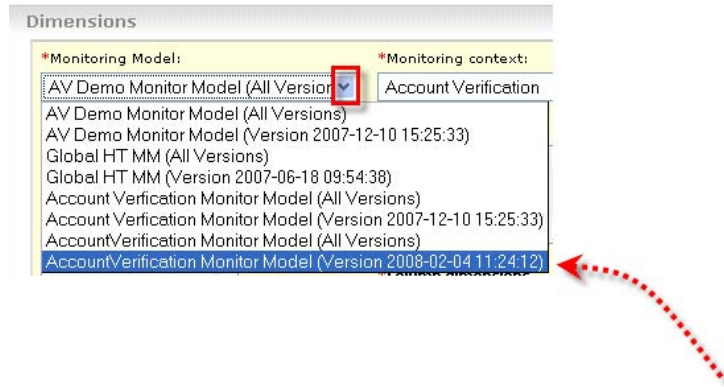
__c. Select an Available Dimension from the left box, then use the left to right > arrow buttons to move the dimensions to the locations they will appear in a chart as shown below. When your selections appear as below, click **OK**.
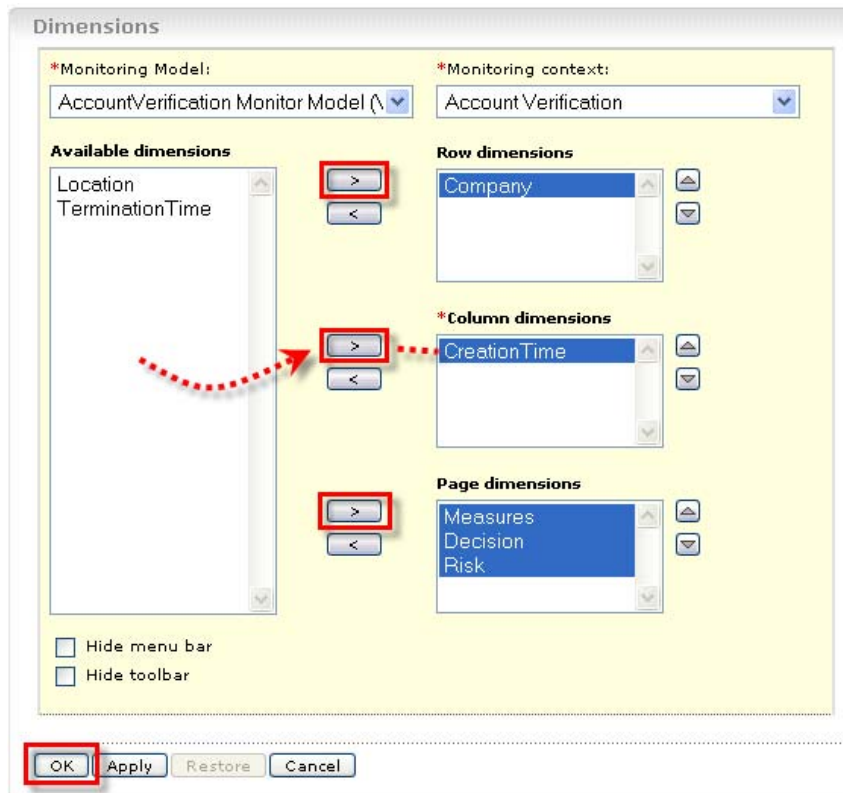
__d. Initially the default chart will appear as below. There is nothing inherently wrong with this chart, but you have decided your audience would benefit more from a pie chart organized by company. You will continue to personalize the chart for that requirement, then Set the final chart as the new default.
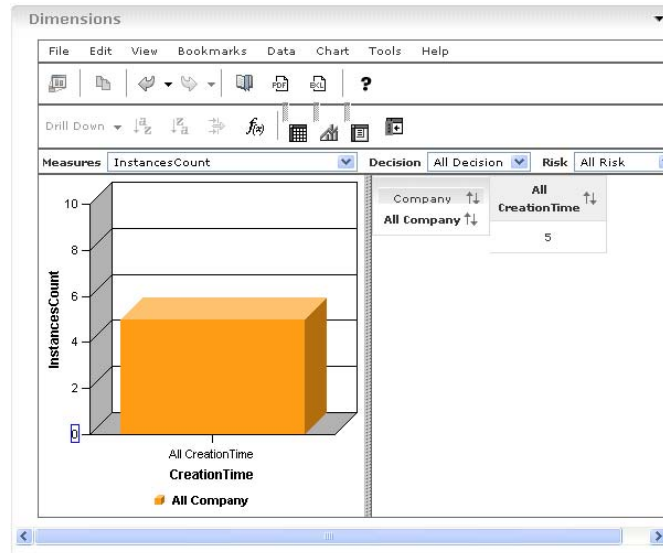
__e. Change the chart type to Pie. On the top line of options in the Dimension view, Click **Chart > Types > Pie.**
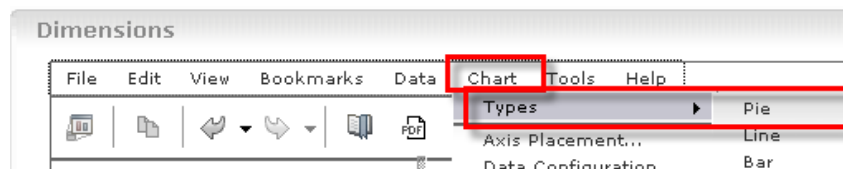
__f. The chart has changed to a pie, but reflects all companies, which totals 100% of the requests. Move the cursor over the pie until a blue rectangle appears, then R**ight click > Drill Down.**
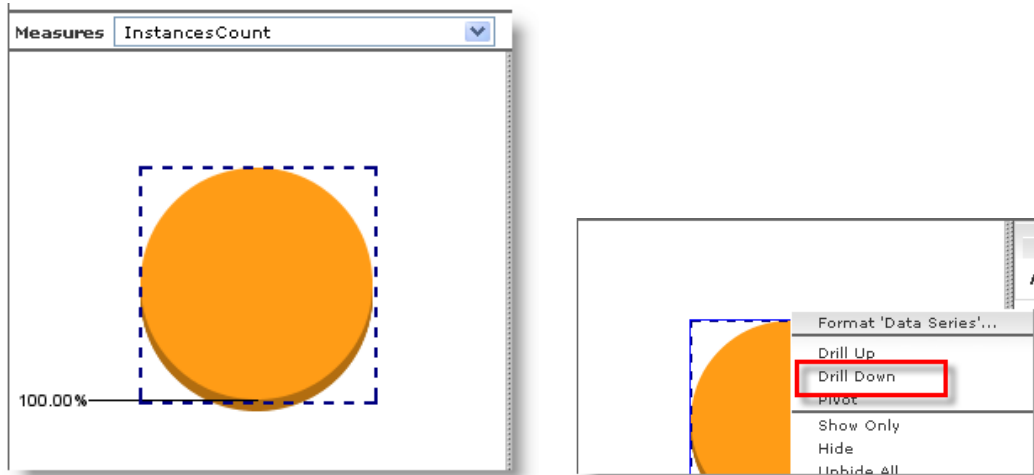
__g. The pie chart is now properly divided by company. As you examine it, you will notice some additional options available in toolbars, and a grid view on the right. These options are not needed to accomplish the objective for our business user, so you can toggle them off.
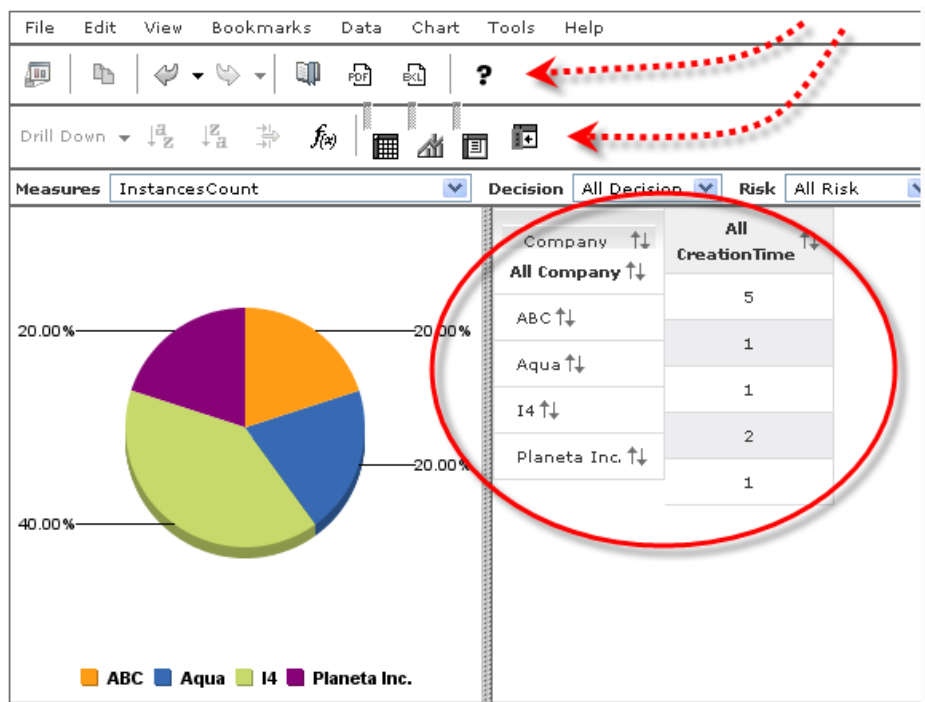
__h. To remove the check marks, turning off the extra toolbars and the grid view:

   __**i.**   Click **View > Toolbar > Standard**

   __**ii.** Click **View > Toolbar > Navigation**

   __**iii.** Click **View > Grid**
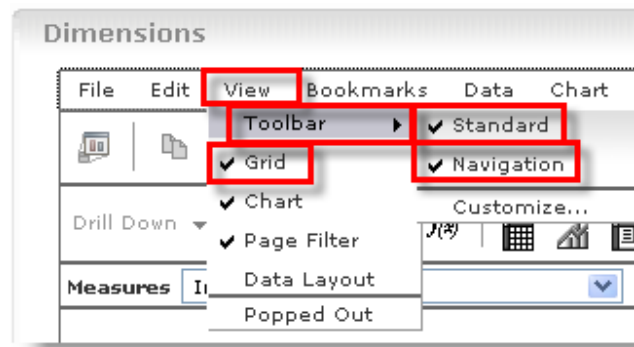
__i. The chart now has a cleaner, simpler appearance. The other options can be toggled back
   on if needed later.
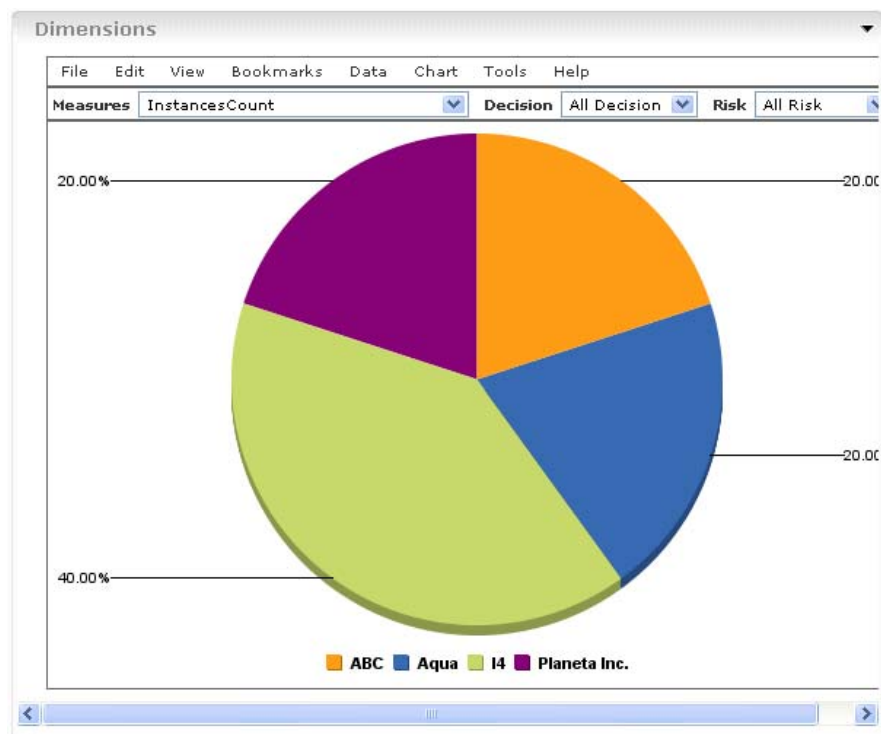
__j. Now that all the chart options are set as needed, click the Set as Default button to ensure the options remain set.
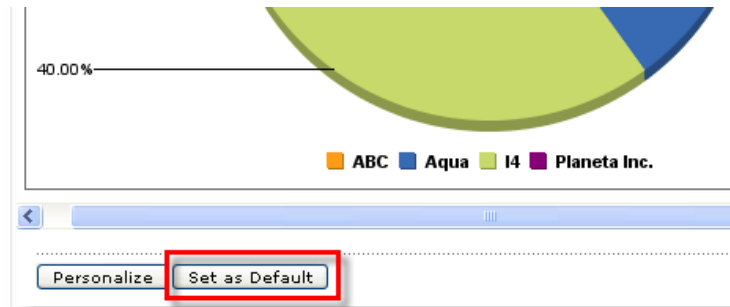
__k. Update the name of the view to **My Dimension View 1.** Click the existing label, enter the new label and click **Save**.
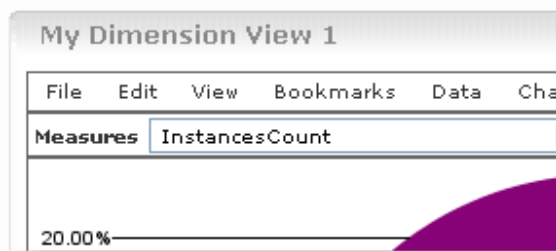
__3. Just as you complete this work, you get a request to add an additional KPI to this dashboard. The user community for this dashboard needs to see a **bar graph** reflecting the **Average Process Cost.** Multiple options are available.

__a. Add a new KPI to My KPI View 1. This means the bar graph wouldn't be possible because different graphic display types (gauges/bars) cannot be mixed in the same view.

__b. Add the new KPI to My KPI View 1, and change the existing gauges to bar graphs. This means the users requesting this original format will not be happy.

__c. Create a new KPI in a new view My KPI View 2, using the bar graph format; this is a better alternative, but it may cause some users (depending on their display resolution) to scroll down to find the new KPI.

__d. Create a new KPI in a new view My KPI View 2, using the bar graph format, and reduce the size needed by MY KPI View 1, by switching the existing full gauges to half gauges.

__4. Any of these options will work; the point is to give this some thought ahead of time, to best organize the types of displays that will be most suitable for the intended audience. For this example, we will choose option d.

__a. Scroll down beneath the My KPI View 1 and click the blue link Add to Dashboard.

___b. Select **KPIs** and click **OK**.
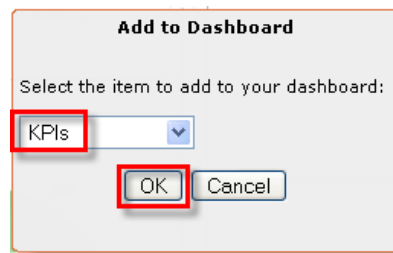
___c. Click the **Personalize** button in this new view.
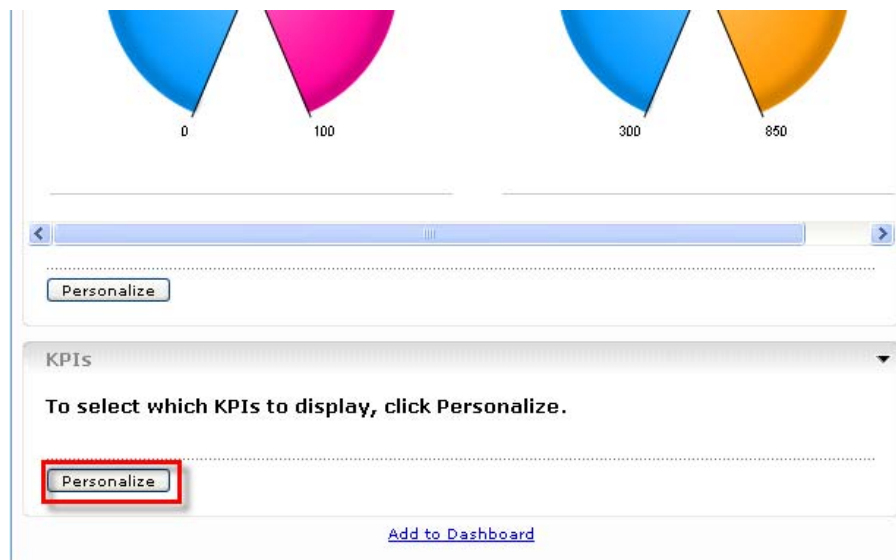
__d. Make sure you select the correct monitor model entry, then place a check by the KPI to be displayed **Avg Process Cost**. Click **OK**.
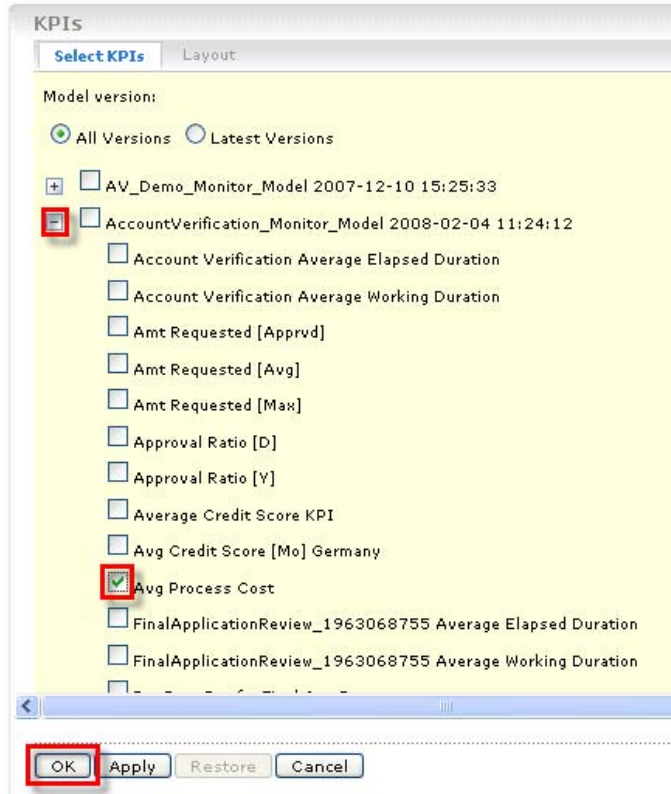
__e. The default KPI display, a table view is shown. To change this to a bar graph, click Personalize again. (you could have set this before you clicked OK)

__f. Click **Layout** and set the display mode to **Bar Graph**, click **OK** at the bottom of the window. The new bar graph is displayed.
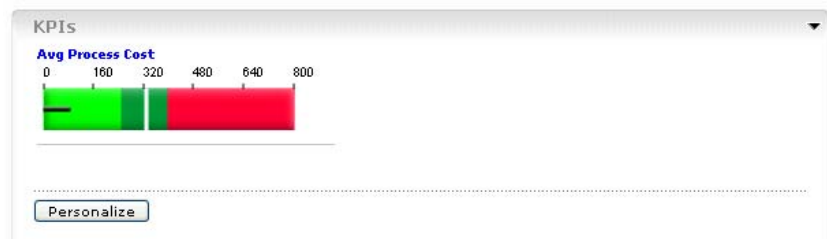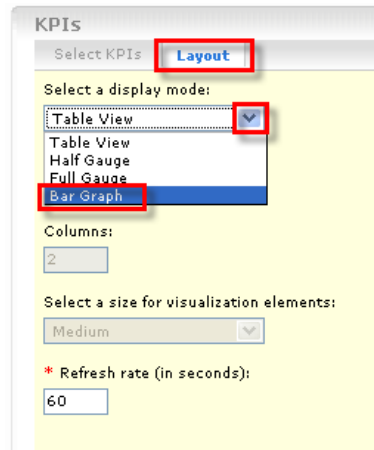
__g. Change the name of the new view from **KPIs** to **My KPI View 2**

__h. Click the Personalize button for the My KPI View 1 window above.

___i. Click **Layout** at the top of the window and change the display mode from **Full Gauge** to **Half Gauge**. Click **OK** at the bottom.
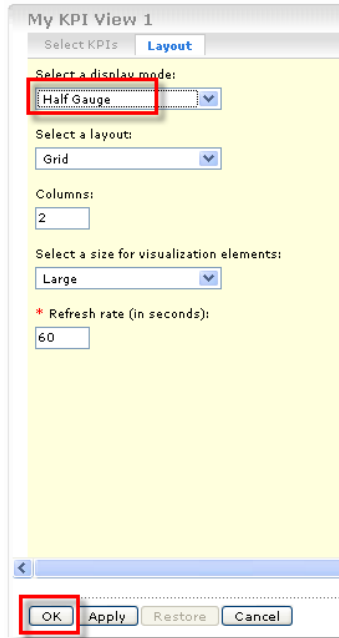
___j. The new modified dashboard should appear as below.
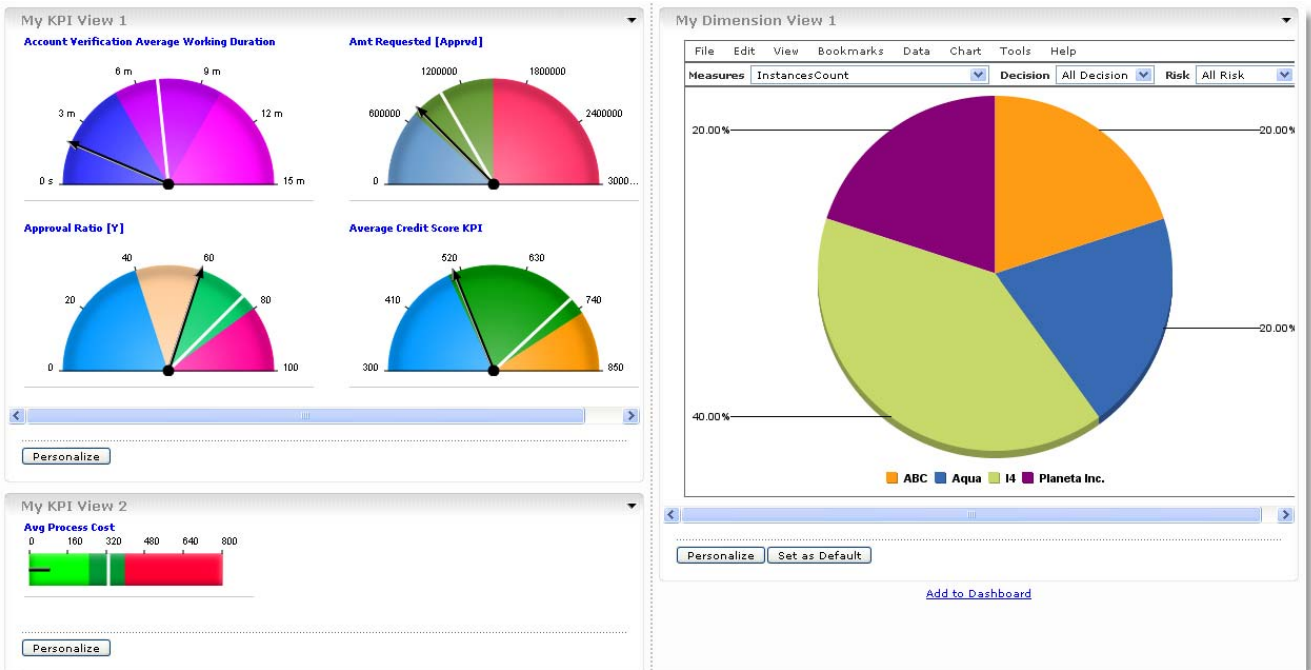
The **Personalize** buttons and Add to Dashboard links are part of the Layout assistance feature. This feature is automatically turned on when you create a new dashboard page. Now that you have completed the layout and personalization, they are not required, and take up space.

__k. Click the Layout Assistance icon at the top right of the page to hide the buttons and links for a cleaner display.
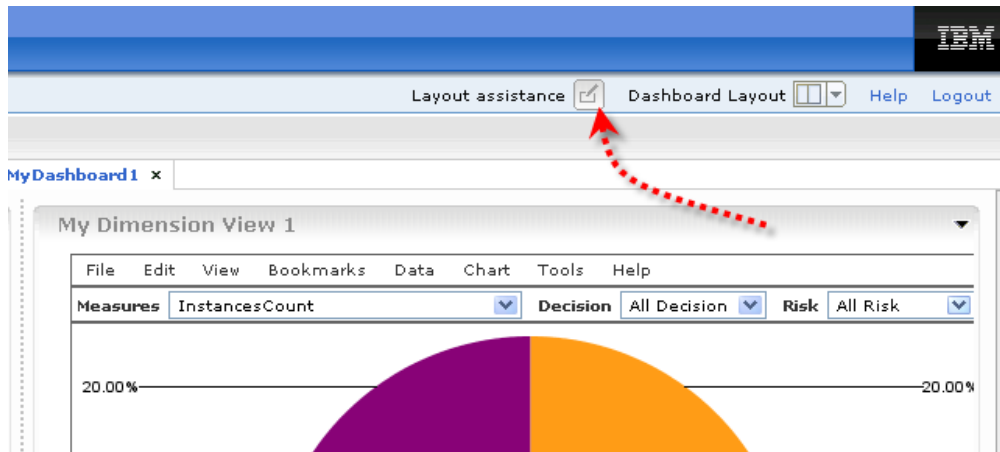
__l. You can turn the layout assistance feature back on by clicking the button above. The Personalize function is also available by clicking the black triangle at the upper right of each view.
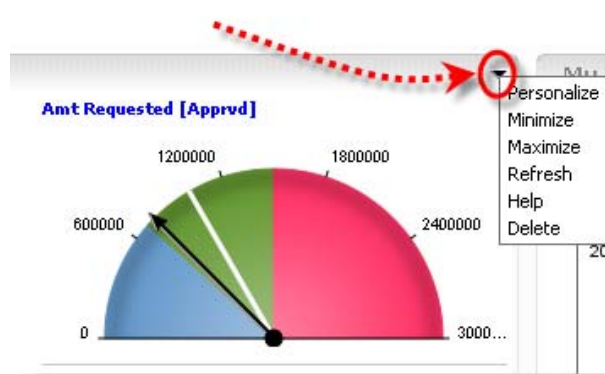
__m. Your completed **MyDashboard 1** page should appear as below:
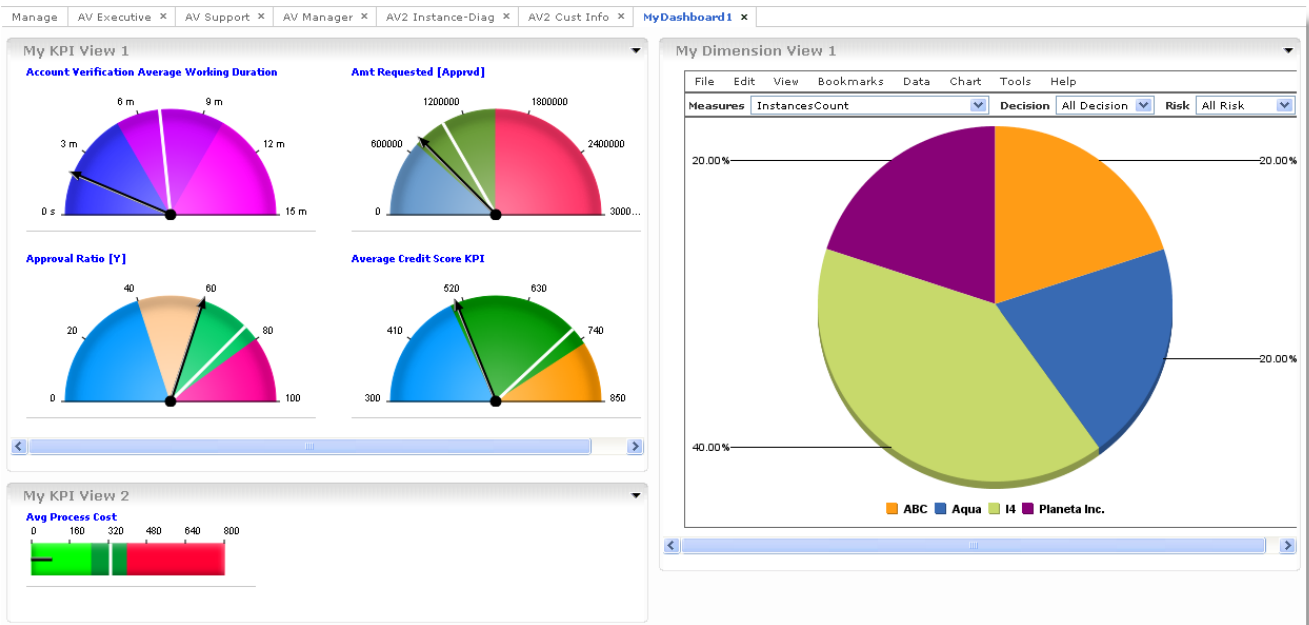
    __n. Before you leave your dashboard page, take some time and explore the ranges, targets and actual values being shown by the gauges and bar graph. Change some selections for Measures, Decision and Risk on the pie and explore.

__5. You have completed this lab.

Lab 15.3 - Run live processes and view with Dashboards

So far, you have been reviewing the details and the measurements from processes that have already started and run to completion. In this lab you will start new instances of the Account Verification process, submitting new requests to the system. To do this, you will use a second browser session to login and assume the role of a Business User Client (such as an account rep) starting these requests on behalf of customers.

The new process instances will actually be running in WebSphere Process Server, which will send events to WebSphere Business Monitor in real time. You will use the dashboards you have been working with to see these new process instances and their effects on the:

___ d.   KPI view (gauges and graphs)

___ e.   Human Task view

___ f.   Process Diagram view

___ g.   Instances view

The diagram below shows a simple view of the setup:

Business User Client

1.   Initiate the process*
2.   Claim human tasks
3.   Complete human tasks

Web Dashboards

1.   Monitor process instances
2.   Monitor process/task measurements
3.   Take limited actions
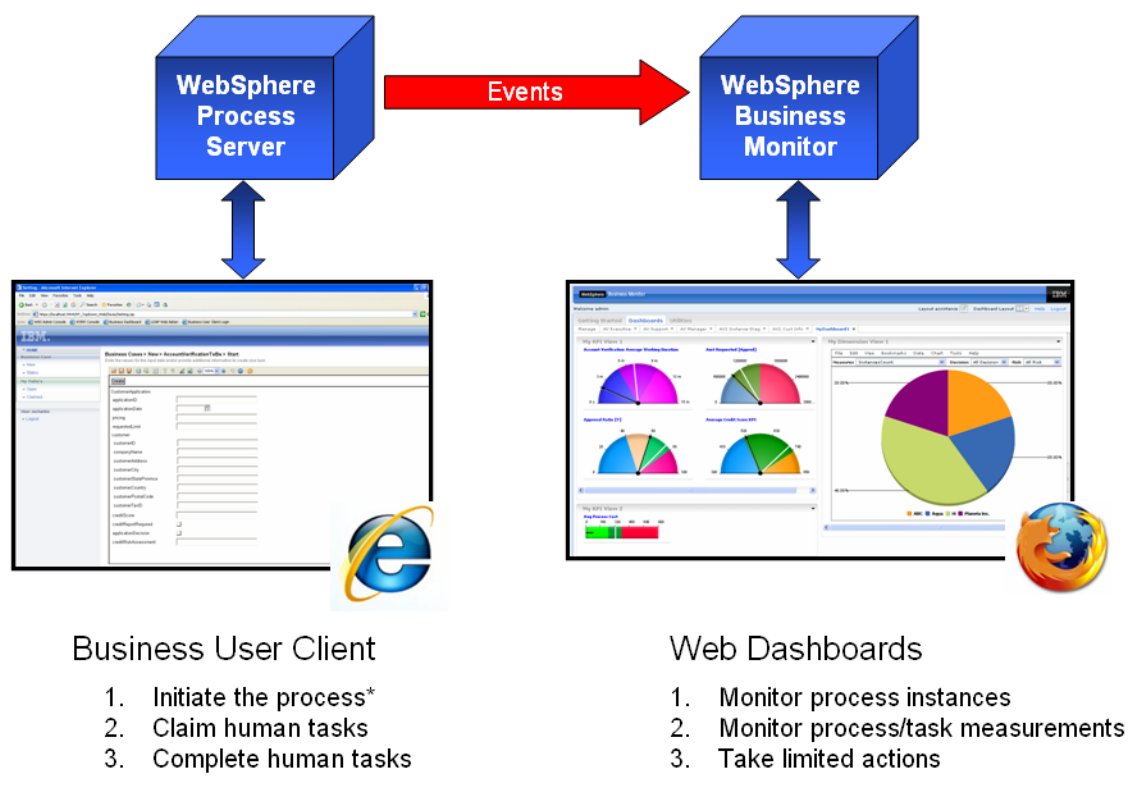
Validate the current state of the processes and measurements

Before sending new requests into the system, in this section you briefly confirm the number and state of the processes recorded in the system, as well as the values for some of the critical business measurements.

As you recall, the views within a dashboard are defined based on a specific model running in the system. To keep things simple, all the dashboards starting with the label AV2, are based on the most recent model, so for this lab, you can reference **any of the AV2 dashboards** in the list, and also the dashboard you created called **MyDashboard1**.

The AV Executive, AV Support and AV Manager dashboards are not linked to the latest model, so their data will not change based on new processes you are about to run.

__1.   Delete AV executive / support / manager from the list ?

__2.   Click the tab to review the **AV2 Instance–Diag** page. Note the 5 instances, all completed.

	Diagram	Account Verification Start Time	Account Verification Elapsed Duration	Application ID	Company Name	Customer City	Customer Country	Amount Requested	Credit Score	Credit Risk	Decision Status	COMPLETED	Account Verification State	Cost - Process	Cost - More Docum
☐		February 24, 2008 6:05:42 PM	1 m, 19.609 s	1203894344515	I4	Munich	Germany	$350,500.00	500	HIGHRISK	Declined	★	3 - STATE_FINISHED	$52.12	$24.77
☐		February 24, 2008 6:04:04 PM	1 m, 28.828 s	1203894244718	Aqua	Melbourne	Australia	$273,000.00	521	HIGHRISK	Approved	★	3 - STATE_FINISHED	$58.81	$32.26
☐		February 24, 2008 6:02:02 PM	1 m, 12.563 s	1203894123062	Planeta Inc.	Tempe	USA	$339,000.00	422	HIGHRISK	Approved	★	3 - STATE_FINISHED	$46.06	$28.47
☐		February 24, 2008 5:59:56 PM	1 m, 49.562 s	1203893998312	I4	Munich	Germany	$140,000.00	500	HIGHRISK	Approved	★	3 - STATE_FINISHED	$72.32	$38.79
☐		February 24, 2008 5:48:53 PM	3 m, 42.657 s	1203893342234	ABC	Thornhill	Canada	$340,000.00	600	MEDRISK	Declined	★	3 - STATE_FINISHED	$172.99	$0.00

|◀ ◀ Page 1 of 1 ▶ ▶| Go to page: [   ] →   Results 1 to 5 of 5

__3.   Use the diagram view at the bottom of this page; scroll over to the **Request More Documentation** and **Final Application Review** tasks. These are the two human tasks in the model. Confirm there are no requests Queued at either of these tasks.
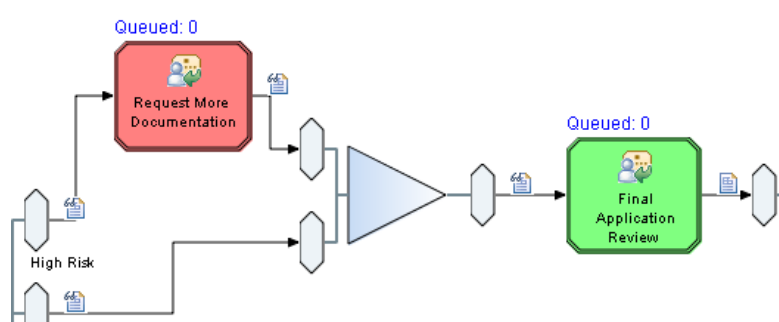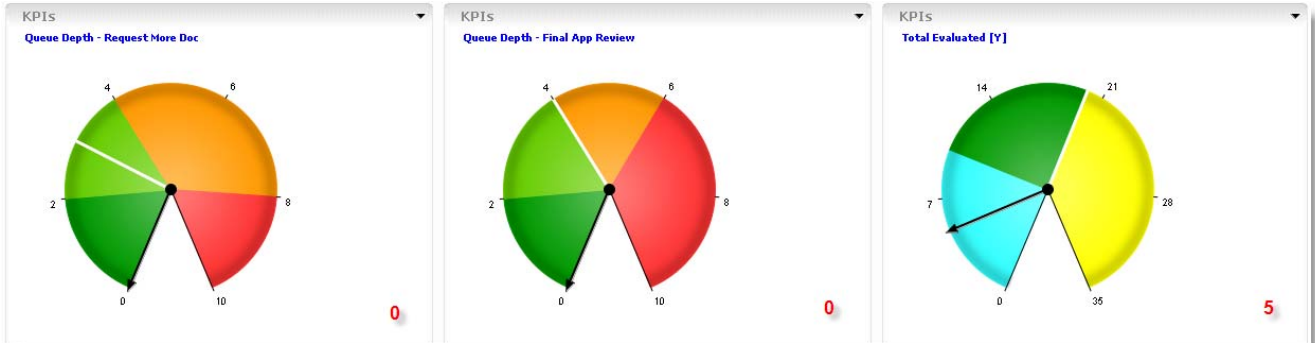
__4.   Click the **Manage** tab, then click the link to review the **AV2 KPI Gauges I** page, the current actuals are shown below in red. You may confirm these by holding the cursor over the black arrows as before.
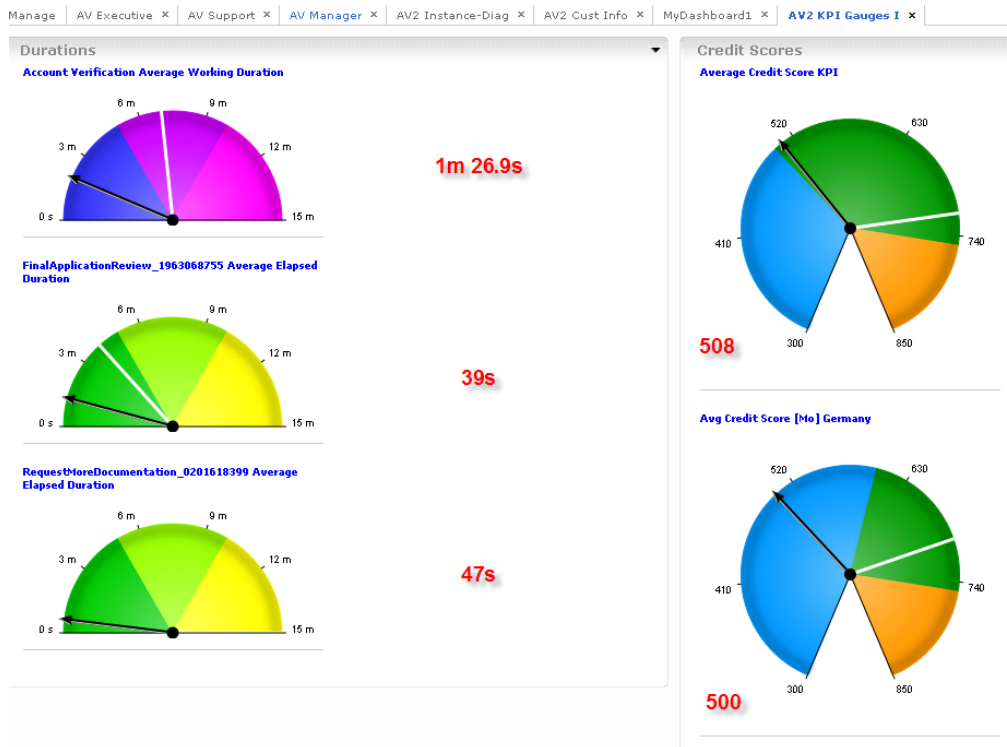

__5.   Click the **Manage** tab, then click the link to review the **AV2 Dim I** page. Notice that requests are shown from a total of 4 companies, with 2 requests from the **I4** company.
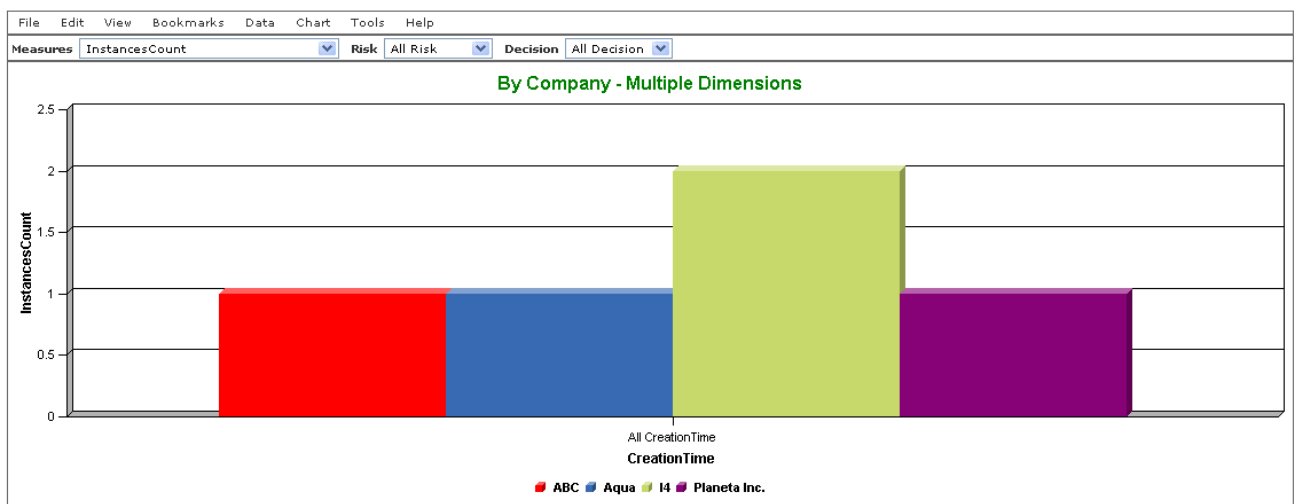
__6. Click the **Manage** tab, then click the link to review the **AV2 KPI Queues** page. Note the KPI's measuring the Queue Depth for both human tasks at zero, and that from the KPI gauge labeled **Total Evaluated [Y]**, a total of 5 requests have been evaluated.

__7. Click the **Manage** tab, then select the link to review the **AV2 KPI Approval** page. The KPI's in the Approval – Daily view are all at zero. In the Annual view at the bottom, the KPI's indicate 3 of 5 requests were approved and the approval ratio is at 60%.
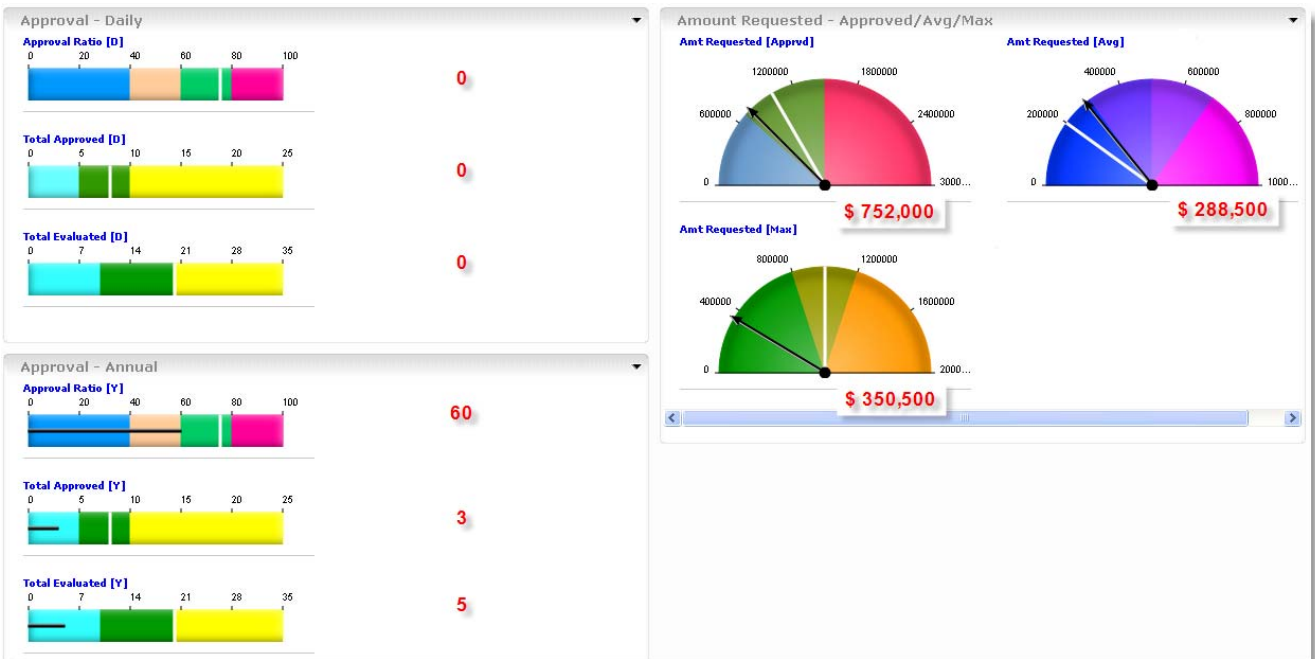
KPI Details

Remember, for any KPI, you can click anywhere on the graphic to show a details page with the description of the KPI

__8.   Click to select the **AV2 Instanced-Diag** dashboard tab.

__9.   Click the '-' in the upper right corner of the browser to Minimize.

__10.   Now that you have reviewed the current performance numbers from the existing processes, you will submit new requests to the system and return to the dashboards to examine the results.

Start New Process instances and view the results

In this section, you will assume another role as a participant in the Account Verification process. You will initiate the process, then the system (WebSphere Process Server) will decide which tasks require your attention. Having reviewed the process diagram, you have seen that there are only two human tasks in the process flow:

- ☐ Request More Documentation

- ☐ Final Application Review

Depending on the credit score and the risk rating of the customer record on file, some requests will complete without requiring any human steps, others will require only the Request for More Documentation, and others will require both human tasks to be completed before the process will end.

The point here is not to focus on the logic used to make the approve or decline decision, but the fact that different process paths are taken, and that these different paths may require human intervention. This human intervention will require claiming the task and perhaps supplying additional information needed by the system. These things will take time, and will impact the overall duration, cost and outcome of the process. It is exactly these factors as well as the general performance indicators we intend to monitor. With WebSphere Business Monitor, this can be done while the processes are running.

__1.   Login as a Business User, to act as a participant in the process.

    __a. From the desktop, double click the Business User Client icon.

    __b. Because of our lab setup, you will receive a security message, click **Yes**.

___c. Use Name: bdaniel   Password: bdaniel and click **Login**.

___d. This is a generic panel that permits interaction with a business process. In production in your environment, this would be a customized panel with company logos and specific functions listed. Note on the left, the logged in user ID is displayed.

___e. To see the list of new processes available which you have permission to start, click **New**.

___2.   Start and run a new process instance, submitting a new customer request to the system.

___a. There is only one process loaded in the system that we have permission to start, the AccountVerification ToBe process. Click **Start** to create and run new process instance.

___b. As the server prepares to start the process, certain information is required to begin the new request. For our setup the **customerTaxID** is the used by the first step in the process to retrieve the remainder of the available customer information for the request.

__c. In production, this information could arrive in any number of ways (electronic requests from the web, phone, e-mail, fax, walk-in etc). In this example, you will supply it manually as a business user of the process.

Also, in this example, WebSphere Process Server is linked to and uses Lotus® Forms to present an electronic form for the business user to interact with.

__d. In the form, click once in the white space to activate the form.

__i. Click once in the box for customerTaxID, and enter a **customerTaxID** of '**2**'.

__ii. Click the **Create** button at the top. This launches the new request based on a customerTaxID, and the retrieval of the customer information will begin.

__e. The process instance is running using the customer information retrieved from customerTaxID '2'.

__3.   Check for open items and claim the work item.

__a. Click Open to display any items waiting for processing.

__b. If no items appear, wait a few seconds, and try the refresh button.

__c. This process has automatically run to the point where the FinalApplicationReview task is required, and is now waiting on human input to complete. Click the **FinalApplicationReview** task to open it.

    __d. Lotus Forms now opens a form representing the customer data retrieved and some additional items that may be needed to complete the process. Click the **Claim** button at the bottom of the panel to claim this work item and begin work.

__4.   Complete the information needed for the Final Application Review.

    __a. Click once somewhere in the form's white space to activate it.

    __b. Click the calendar button and choose an applicationDate. The current date is ok.

    __c. Enter **Select** for pricing

    __d. Enter **275000** for requestedLimit

    __e. Note the applicationDecision box:

        __i.   if this box is checked, the customer request is Approved

     __ii.   if it is unchecked, the customer request is Declined

     __iii.   leave the box unchecked for this request

  __f. Click the **Complete** button at the top of the form when you are done.

__5.   The display will refresh and you should now see: **Items found: 0** There are no additional items waiting on your input at this time.

__6.   Hold the **Alt** key and press the **Tab** key to switch to the Business Dashboard session in the other browser. Alternatively, you may choose to minimize the current browser (IE – Business User Client) then click the button on the toolbar for the Business Dashboard session (Firefox browser) if you prefer.

__7.   You should see the **AV2 Instance-Diag** dashboard with your new process instance now visible.

☐	🔧	February 24, 2008 7:36:13 PM	7 m, 52.828 s	1203899776265	ABC	Thornhill	Canada	$275,000.00	600	MEDRISK	Declined ★	3 - STATE_FINISHED $352.98 $0.00
☐	🔧	February 24, 2008 6:05:42 PM	1 m, 19.609 s	1203894344515	I4	Munich	Germany	$350,500.00	500	HIGHRISK	Declined ★	3 - STATE_FINISHED $52.12 $24.77
☐	🔧	February 24, 2008 6:04:04 PM	1 m, 28.828 s	1203894244718	Aqua	Melbourne	Australia	$273,000.00	521	HIGHRISK	Approved ★	3 - STATE_FINISHED $58.81 $32.26
☐	🔧	February 24, 2008 6:02:02 PM	1 m, 12.563 s	1203894123062	Planeta Inc.	Tempe	USA	$339,000.00	422	HIGHRISK	Approved ★	3 - STATE_FINISHED $46.06 $28.47
☐	🔧	February 24, 2008 5:59:56 PM	1 m, 49.562 s	1203893998312	I4	Munich	Germany	$140,000.00	500	HIGHRISK	Approved ★	3 - STATE_FINISHED $72.32 $38.79
☐	🔧	February 24, 2008 5:48:53 PM	3 m, 42.657 s	1203893342234	ABC	Thornhill	Canada	$340,000.00	600	MEDRISK	Declined ★	3 - STATE_FINISHED $172.99 $0.00

|◀ ◀ Page 1 of 1 ▶ ▶| Go to page: [   ] ↵   Results 1 to 6 of 6

__8.   Click the tab to review the **AV2 KPI Approval Info** page. Refer to page 154 for the earlier values.

  __a. Amt Requested [Avg]: **$ 286,250** down from $ 288,500 (lowered by your $275,000 request)

  __b. In the Approval – Daily view, you should now see a value in the bar graph for **Total Evaluated [D]**, but since you did not approve the request, the **Approval Ratio [D]** is zero and the **Total Approved [D]** is zero.

  __c. Approval Ratio [Y]: **50%**, down from 60%

  __d. Total Approved [Y]: unchanged from 3, as you decided not to approve the request.

  __e. Total Evaluated [Y]: **6**, up from 5

__9.   Click the tab to review the **AV2 KPI Gauges I** page. Refer to page 153 to check the earlier values.

  __a. The Account Verification Average Working Duration will likely be higher, depending on how long you took to perform the steps. The earlier value was 1m 26.9s

  __b. The FinalApplicationRevew… Duration may also be higher. The earlier value was 39s.

  __c. The RequestMoreDocumentation…Duration will be unchanged, because the request you submitted did not require this step.

  __d. The Average Credit Score KPI: 523, up from 508

  __e. Click the other tabs and explore differences you may notice.

__10.   Submit a new request.

  __a. Hold the **Alt** key and press the **Tab** key to switch back to the Business User Client. Submit another new request.

  __b. Click **New**, Click **Start**
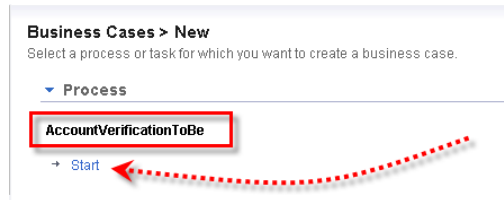
__c. Click white space in the form to activate it.

__d. Click once in the customerTaxID box

__e. Enter **3** for the customerTaxID

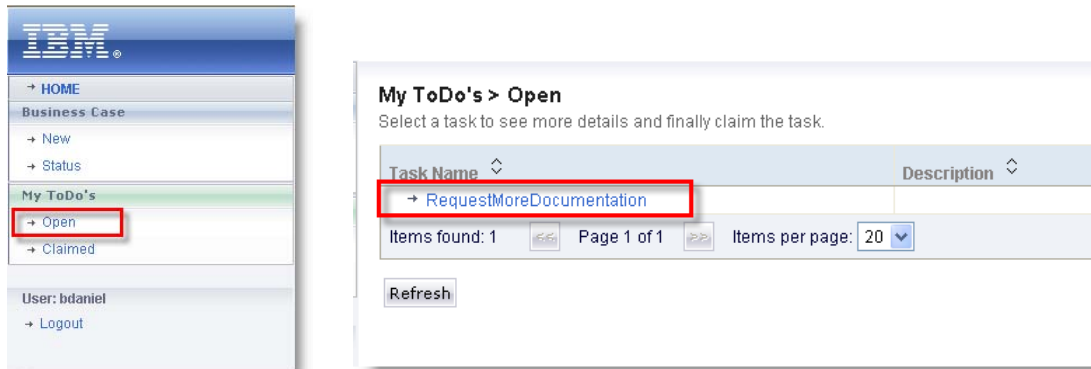__f. Click the **Create** button at the top of the form.

__g. Click **Open** to see any waiting items. Click **RequestMoreDocumentation**.

(if no item appears, wait a few seconds and click the refresh button)

__h. Click the **Claim** button to assign the work item to you, and mark the beginning of work on this task.
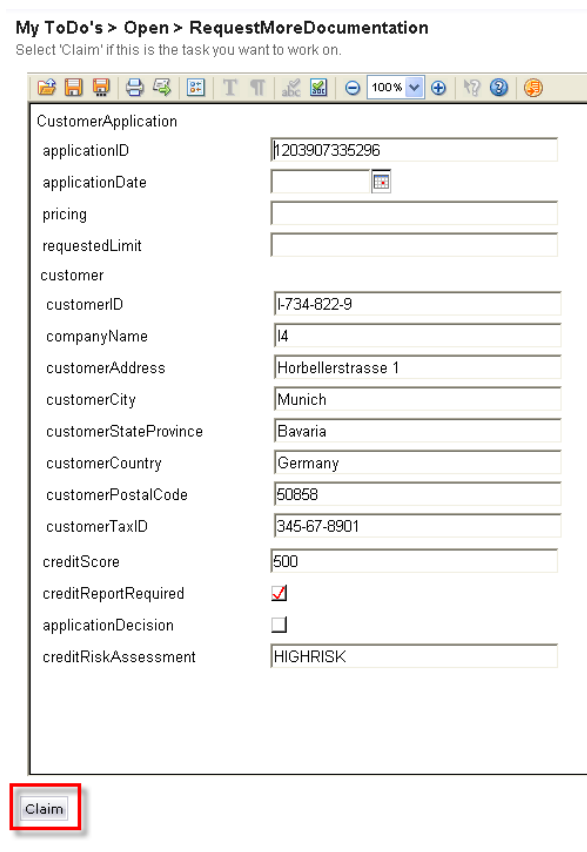
__i. You will now pause work on this task, and check to see what has changed from the perspective of the Dashboards.

__11.   Switch to the dashboards and check for updates

__a. Hold the Alt key and press the Tab key to switch sessions to the dashboards again.

__b. The **AV2 Instance-Diag** page should now reflect the new instance, not yet completed with a **Decision Status** of Pending, and a state of **RUNNING**.

	Diagram	Account Verification Start Time	Account Verification Elapsed Duration	Application ID	Company Name	Customer City	Customer Country	Amount Requested	Credit Score	Credit Risk	Decision Status	COMPLETED	Account Verification State
☐	🔧	February 24, 2008 9:42:14 PM	5 m, 33 s	1203907335296	I4	Munich	Germany	$0.00	500	HIGHRISK	Pending		2 – STATE_RUNNING

__c. Scroll down so that the Process Diagram is displayed, scroll over to the Request More Documentation task that you have claimed. It should show **Queued: 1. T**his indicates that for all instances, only one request is active at the Request More Documentation step.
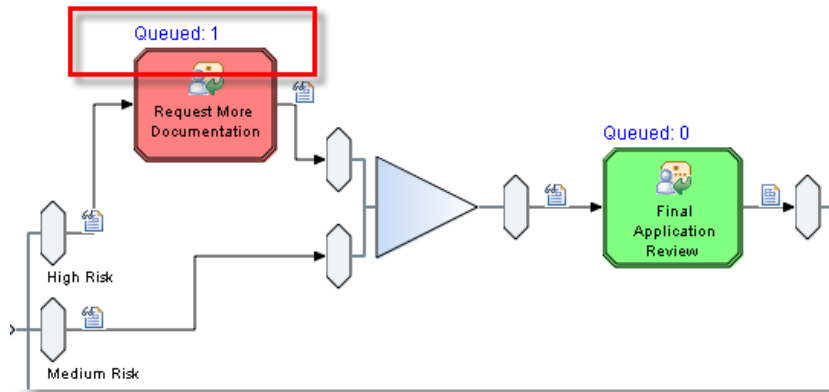
__d. For your new process instance, click the Diagram icon in the on the left. This will refresh the process diagram with information only from this one request.

	Diagram	Account Verification Start Time	Account Verification Elapsed Duration	Application ID	Company Name	Customer City	Customer Country	Amount Requested	Credit Score	Credit Risk	Decision Status	COMPLETED	Account Verification State
☐	🔧	February 24, 2008 9:42:14 PM	6 m, 35 s	1203907335296	I4	Munich	Germany	$0.00	500	HIGHRISK	Pending		2 – STATE_RUNNING
☐	🔧	February 24, 2008 7:36:13 PM	7 m, 52.828 s	1203899776265	ABC	Thornhill	Canada	$275,000.00	600	MEDRISK	Declined	★	3 – STATE_FINISHED
☐	🔧	February 24, 2008 6:05:42 PM	1 m, 19.609 s	1203894344515	I4	Munich	Germany	$350,500.00	500	HIGHRISK	Declined	★	3 – STATE_FINISHED

__e. Scroll across the refreshed diagram, the completed tasks are colored in blue, the Request More Documentation task is highlighted in red, indicating this step is actively processing, and that the customer request will proceed only when this step is completed.
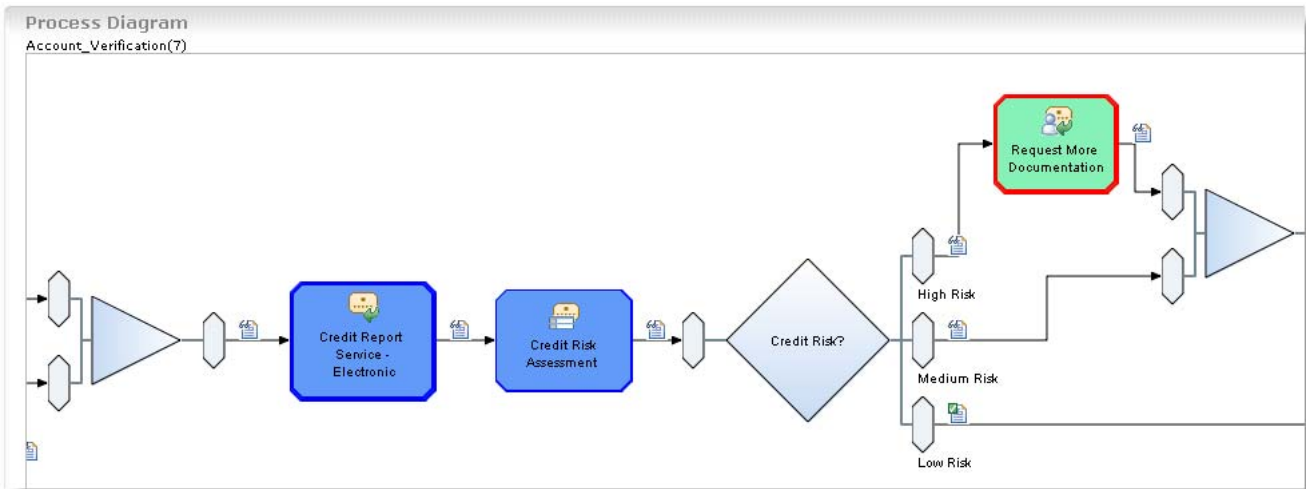
__f. Click the tab for the **AV2 KPI Queue** dashboard. You will notice the **Queue Depth – Request More Doc** gauge has now gone up to a value of one.
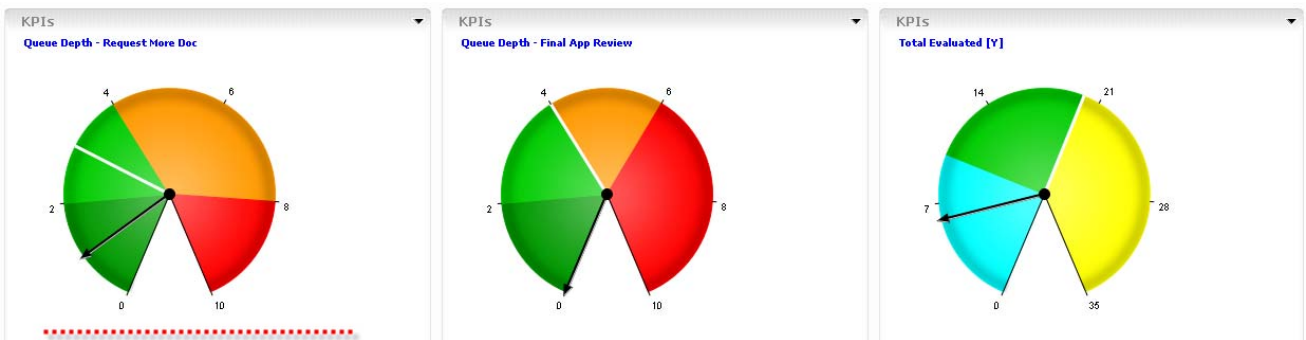
__12.	Switch back to the Business User Client and complete the claimed **Request More Documentation** task.

	__a. Hold the **Alt** key and press the **Tab** key to switch sessions.

	__b. Click the white space of the form to activate it.

	__c. Supply a date, **Elite** pricing and a requested limit of **390000**.

	__d. Click the **Complete** button.

__e. Click the Refresh button, the **FinalApplicationReview** step should appear.

__f. **Claim** the Final Application Review task.

__g. Check the box to approve the **applicationDecision**.

__h. Click the **Complete** box at the top of the form.

__13. Use **Alt+Tab** to switch to the dashboard session.

    __a. Check the **AV2 Instance-Diag** page for your request. It should now show a Decision Status of **Approved** and a State of **FINISHED**.

    __b. Check the other **AV2** dashboard pages for other changes you may notice.

        Note: In this configuration, cost is calculated on a per second basis for each task, so the longer it takes to perform the two human tasks the costs for those tasks, and therefore the process cost will go up.

        In this section you have started new process instances and looked at some of the resulting changes in the dashboards provided by WebSphere Business Monitor. You have seen that WebSphere Business Monitor is a near real time monitor. In this configuration, new process updates are seen in the dashboards about one minute after processing on the server. You have seen that the KPIs, diagrams and instance views in the dashboards are updated while the processes are still in flight, especially important for monitoring long running processes.

Lab 15.5 - Optional: Start multiple new processes and view results

In this optional section several new requests will be started using different user logins, this will provide another dimension for analysis on one of the dashboards. Also multiples of each human task will be claimed, so that multiple queued items will be reflected in the process diagram view, and the AV2 KPI Queues gauges.

__1.   Perform the following steps from the Business User Client to start several new requests. (use Alt+Tab to switch sessions)

   __a. You are already logged in as bdaniel, click **Logout** and start over with another ID.

   __b. **Login** with an ID; passwords are identical to the user ID

      ___ t.   redwards

      ___ u.   mcharles

      ___ v.   bdaniel

      ___ w.   admin

   __c. Click **New**, Click **Start**

   __d. Supply a **customerTaxID** in the form.   (suggest using different taxID's each time)

      ___a.   Use these tax ID's: 1, 2, 3, 5, 6, 7, 8, 9, 10, 11   (taxID 4 may cause problems)

   __e. Click **Create** at the top of the form   (process runs until a human task is encountered)

   __f. Click **Open** to check for waiting tasks

   __g. Click the task name, wait for the form, then **Claim** the task

   __h. Logout, the go to step b. and go again with another user ID. When you have done the above a few times, proceed.

__2.   Switch to the dashboards:

   __a. If you used different taxID's for your requests, the AV2 Dim I dashboard page will confirm several more companies now with requests.
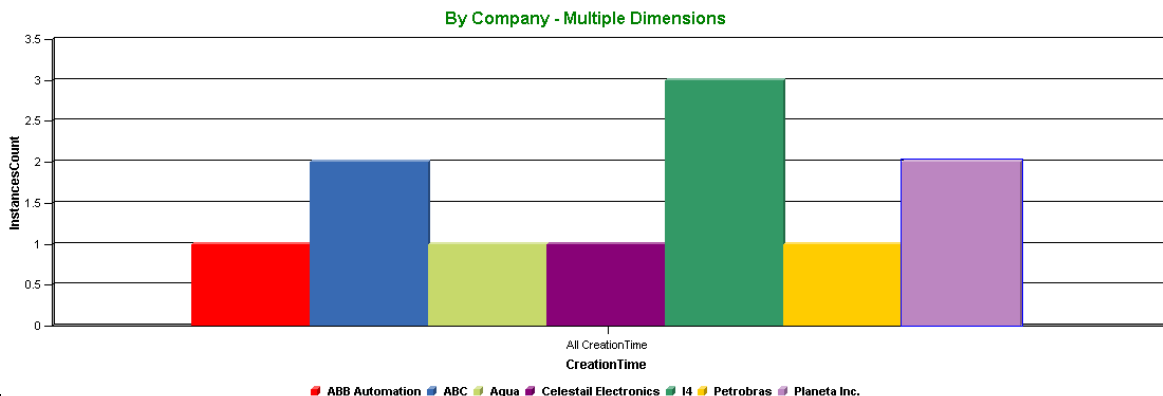
__b. The AV2 KPI Queues dashboard indicates that 4 new requests are queued for Request More Doc Tasks. (in this example)
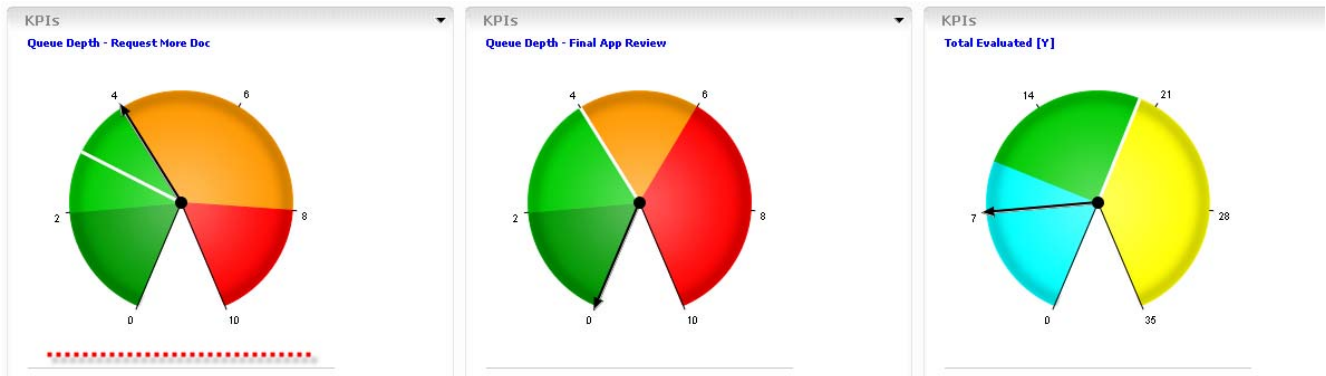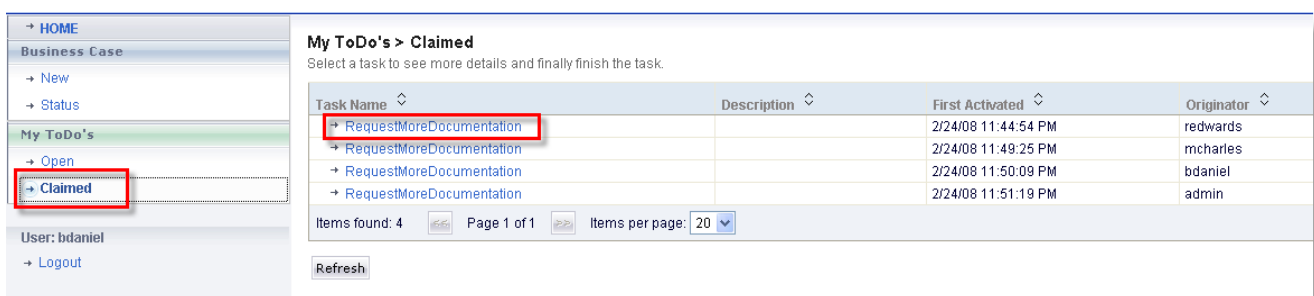

__3. Complete some of your existing claimed tasks.

__a. Switch back to the Business User Client session.

__b. Login with a userid who originated one of the other tasks

__c. Click **Claimed**, then click one of the tasks *originated by the ID you are using*.

__d. Click the calendar to provide an applicationDate

__e. Enter pricing of Select, Priority or Elite

__f. Supply a requestedLimit. 10000 – 1000000

__g. Supply a creditScore if it is not present (350-850)

__h. Click **Complete**.

__i. **Logout** and go back to step b – repeat until you have 2 RequestMoreDocumentation tasks left in that list.

__4. Claim some of the FinalApplicationReview tasks now waiting:

__a. Click **Open**, then click on of the FinalApplicationReview tasks

__b. **Claim** the task   (do not complete it)

__c. **Logout,** use another ID to login and repeat

__5.   Switch back to the dashboards and review the results.

__a. The **AV2 KPI Queues** page shows multiple of both human tasks are queued.
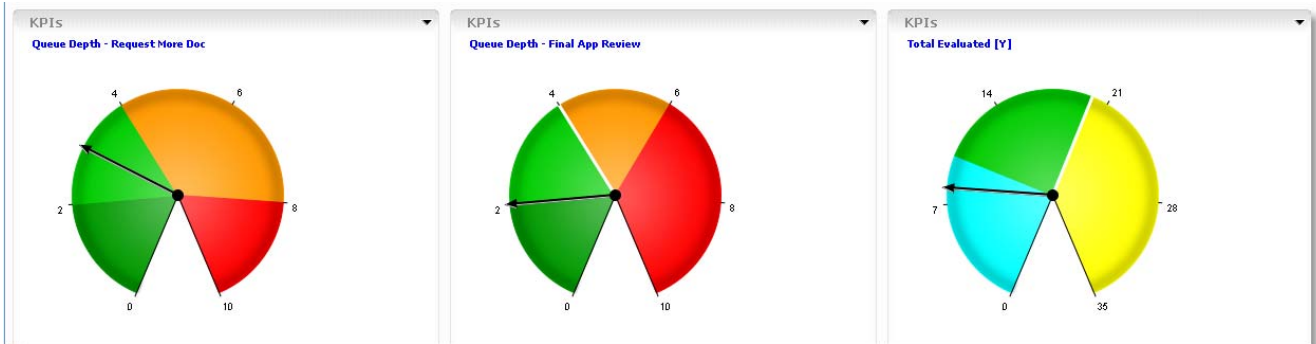
__b. The Process Diagram view in the **AV2 Instance-Diag** page indicates where the work is currently occurring in the processes, confirming the gauges from the earlier views.
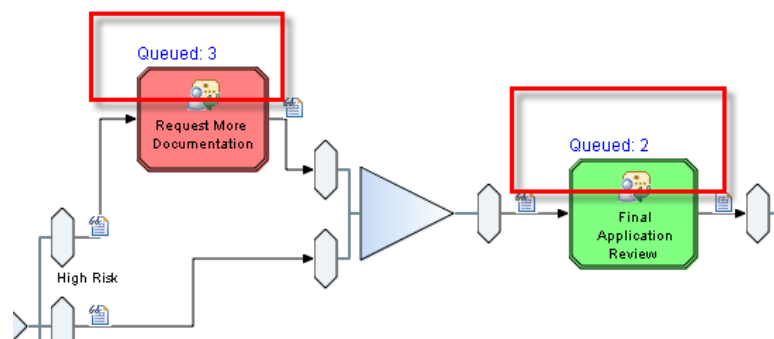

__c. Click the Diagram icon for some of your new requests on the **AV2 Instances-Diag** page. The refreshed process diagram shows which specific task is waiting to be completed. For some of your new instances, it will be Request More Documentation task. For others, the Final Application Review will be awaiting completion.
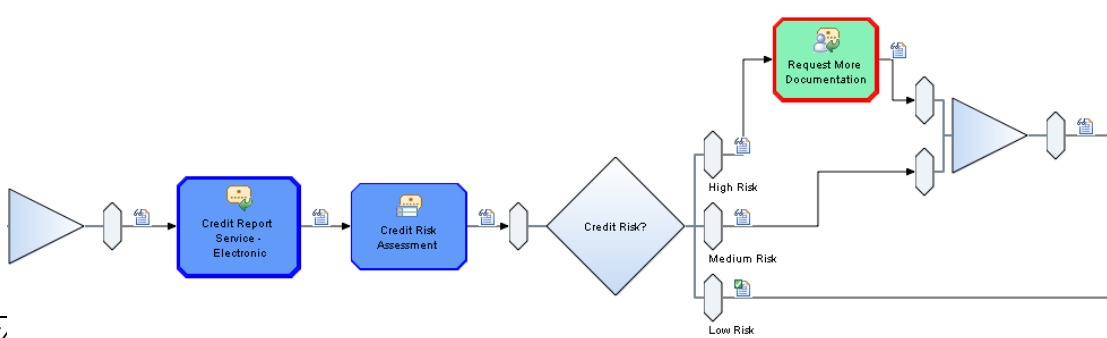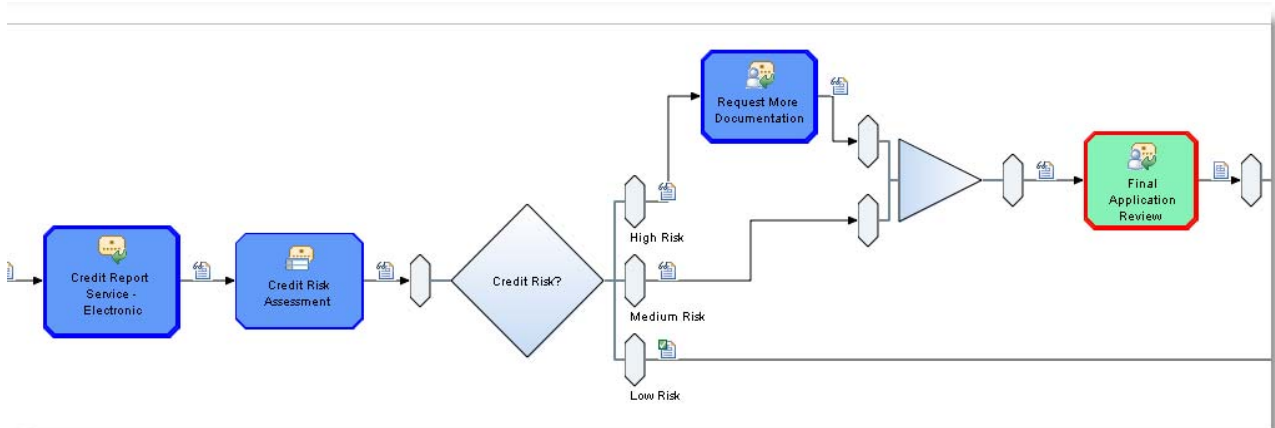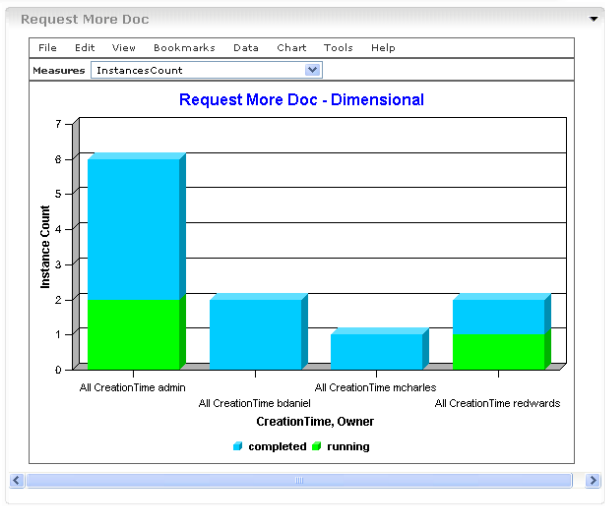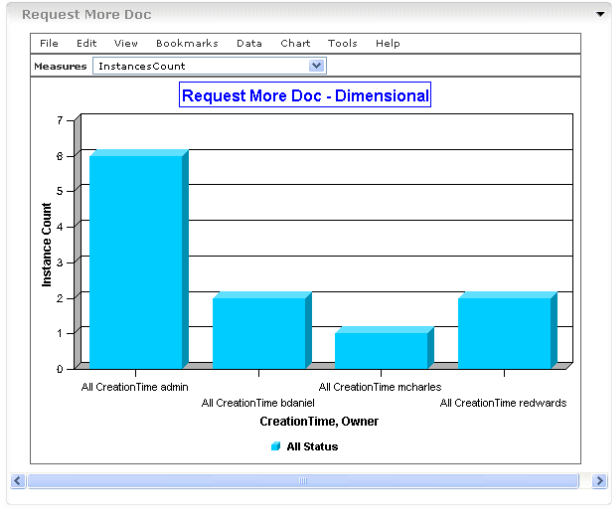
__d. The **MyDashboard1** page indicates a number of new companies making requests.
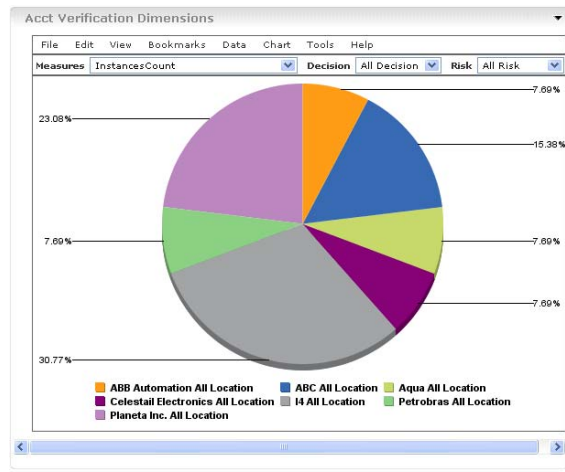
__e. From the **AV2 HT Dim** dashboard page, the **Request More Doc – Dimensional** view on shows a count of these tasks by owner. If you drill down by **Status**, you can see a color coded display of **Completed** vs. **Running** counts for this task, by owner.

__f. Also from the **AV2 HT Dim** dashboard page, using a different display configuration, the **Final App Review** panel displays two groups of columns by status, with individual columns by owner.
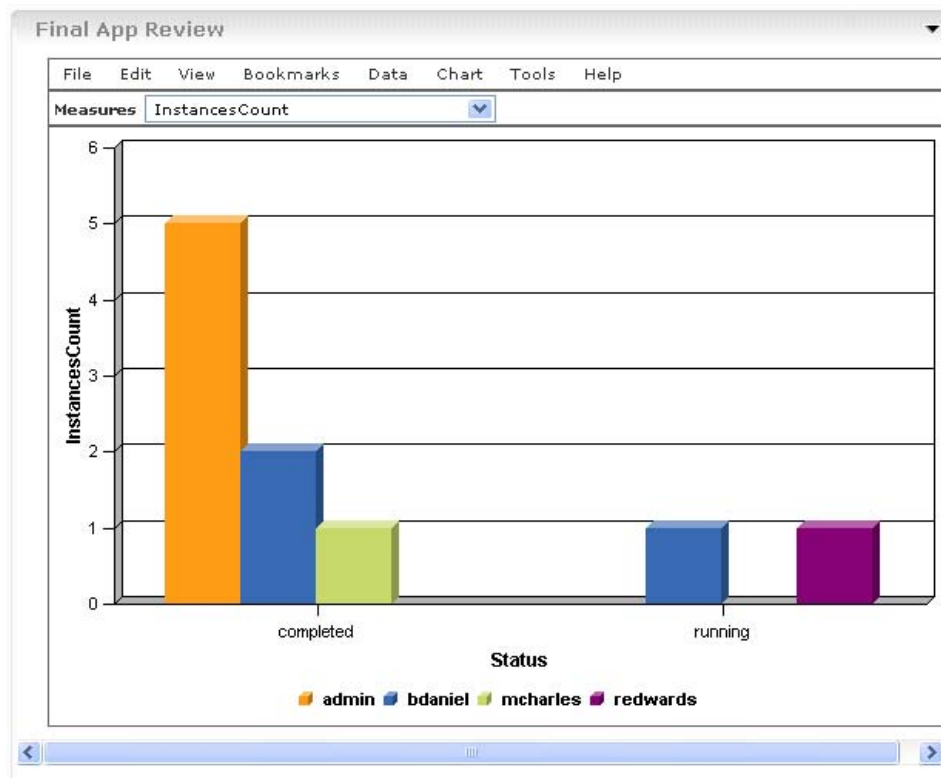
__6.   Optional: At this point, you may choose to switch back to the Business User Client and complete the running processes. If you choose not to do this, and you revisit the dashboards later, the elapsed durations may well be at the edge of their scale. Otherwise, leaving these processes running should not cause any problems.

    __a. Switch (use ALT+TAB) to the Business User Client

    __b. The idea here is to process any tasks which are open or claimed until there are none left.

    __c. Whether you've clicked on Open or Claimed:

        __i.   Click the task name, (if Open, Claim the task)

        __ii.   Supply the information described earlier (page 173)

        __iii. Complete the task

__7.   Click **Logout** from the Business User Client and close the Internet Explorer browser.

Create new KPIs; Alert Subscription; Export Actuals

In this lab, you will look briefly at some additional capabilities of WebSphere Business Monitor. New KPI's can be created dynamically by business analysts using the dashboards. Business users can also subscribe and remove their subscriptions to the alerts that have been defined in the system. The collected data from the actual process instances stored in the monitor database can be used to assist process improvement by exporting a subset of this information and making it available to WebSphere Business Modeler.

Create new KPI's

In the earlier labs, you used KPI's already defined in the existing models and viewed them as they were configured on various dashboards. You also added some of these KPI's to your own new dashboard. Now as a business user, you will create an entirely new KPI and add it to your own dashboard.
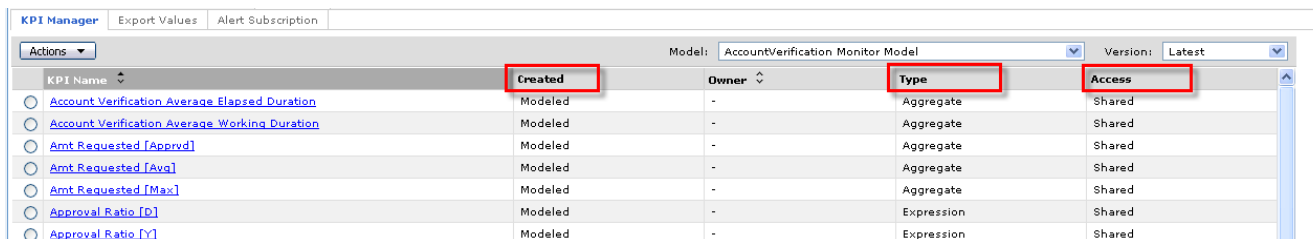
__1. From the Dashboards tab, click the **Manage** tab and click the link for the **MyDashboard1** page. If a tab for **MyDashboard1** is still available on your display, just click it directly if you wish. Your graphs, and gauges values will be different now, depending on earlier lab steps you completed.
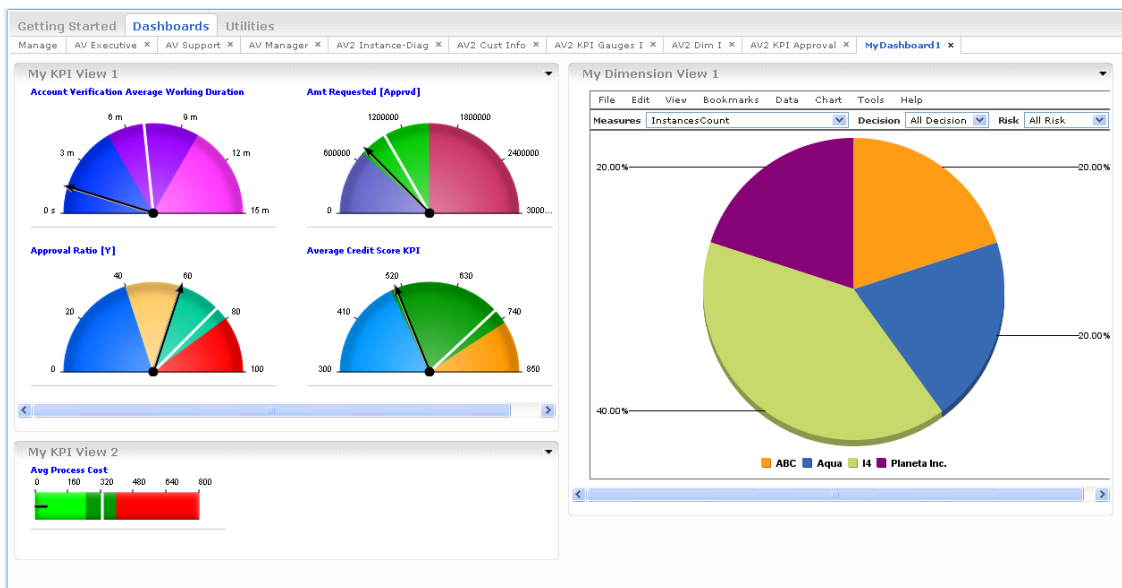
__2.   The business community has asked for a separate view where the minimum credit amount requested could be displayed. The new KPI will be called **Amt Requested [Min]**.

__3.   Click the **Utilities** tab, located to the right of the Dashboards tab. The KPI Manager page will appear, as one of the utilities. Use the drop down arrow to the right, and select the **AccountVerification Monitor Model**.
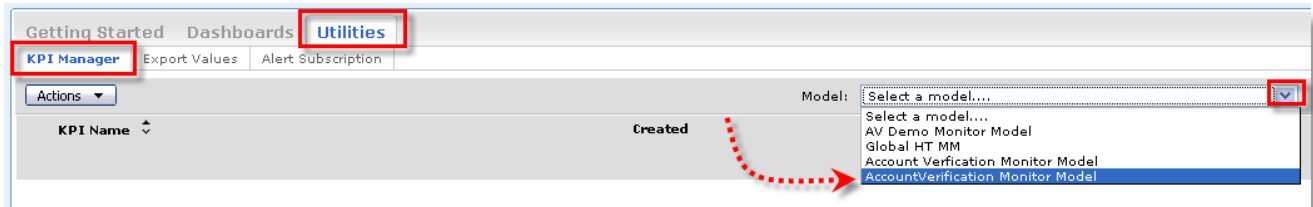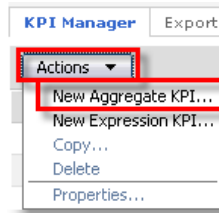

__4.   A list of the defined KPI's will appear. Note the column headings:

   __a. **Created**:

      __i.   **Modeled**: indicates the KPI was created in the Model now running on the server

      __ii.   **Dashboard:** the KPI was created by a Dashboard user, after a model was deployed. You are about to create a Dashboard KPI. These are also referred to as Dynamic KPI's, since they are created "in flight", while the monitor is running.

   __b. **Type**:

      __i.   **Aggregate**: the KPI uses a Metric (such as process cost or amount requested) combined with an Aggregation function such as Average, Count, Sum, Min/Max). e.g. Process Cost [Avg] or Total Approved [D]. Your new KPI will be of this type.

      __ii.   **Expression**: the KPI uses an expression built by the user referencing KPI's and mathematical operators or user-defined functions to calculate the value of the KPI

   __c. **Access**:

      __i.   **Shared**: other dashboard users can use the KPI

      __ii.   **Personal**: other dashboard users cannot view the KPI

__5.   To create the new KPI, click **Actions** the select **New Aggregate KPI**.
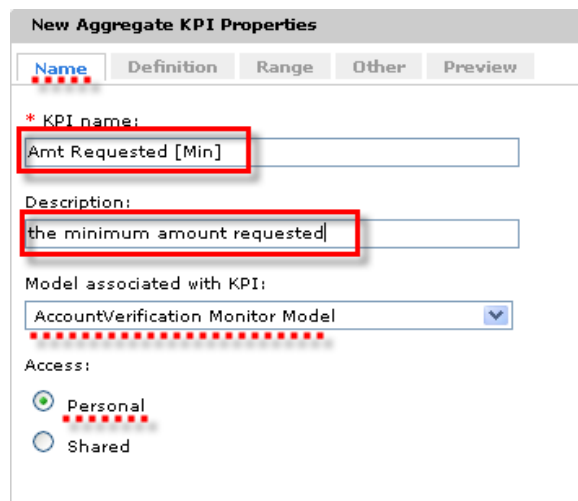

__6.   A KPI properties panel will appear; the **Name** tab will be visible.

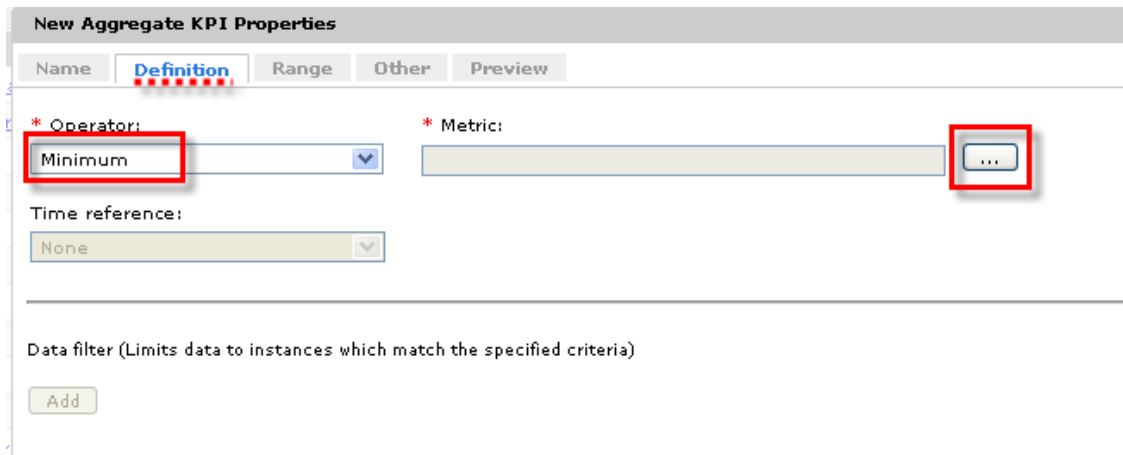__a. Enter the KPI name: **Amt Requested [Min]**

__b. Enter a description: **the minimum amount requested**

__c. Model associate with KPI should be: **AccountVerification Monitor Model**

__d. Access can be left as Personal.

__7.   Click the **Definition** tab.

      __a. Select **Minimum** for the operator.

      __b. Click the Ellipsis button to the right of the Metric box.
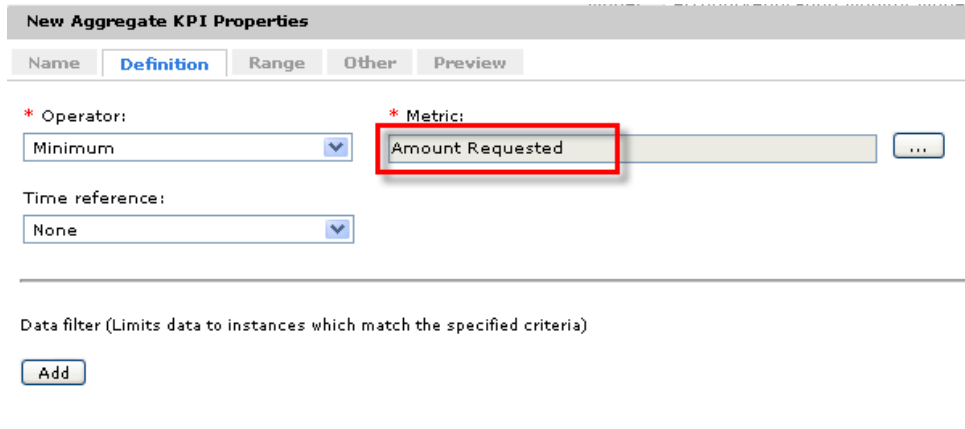
__8.   Click the '**+**' to expand the Account Verification tree, select **Amount Requested** and click **OK**. You are selecting an Instance Metric here, which is a value captured for every request sent to the system. These instance metrics were a part of the original model. You are leveraging them in another way when you create new KPI's that use them.
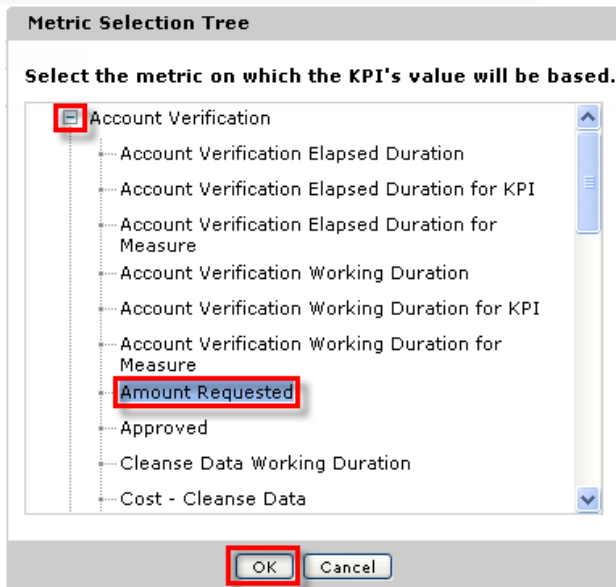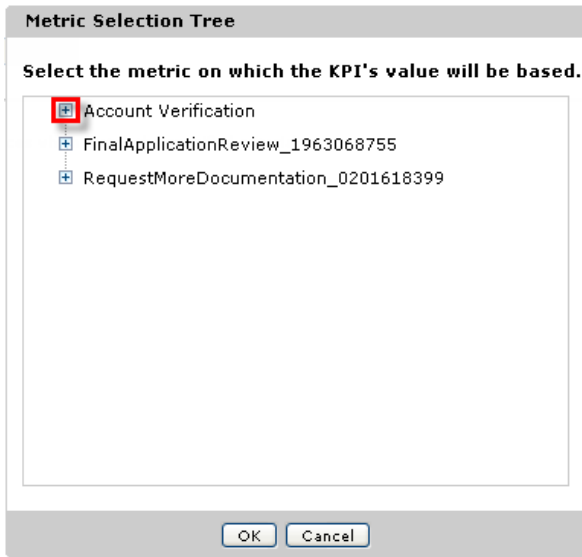
__9.   The **Amount Requested** metric is now defined as the source where your new KPI will get its values. The Time reference and Data filter are mechanisms you could use to limit the information displayed to certain time windows (daily, weekly etc.) or any conditions we want to place on this KPI. (limit this KPI to where Region = North or Customer Type = Gold)
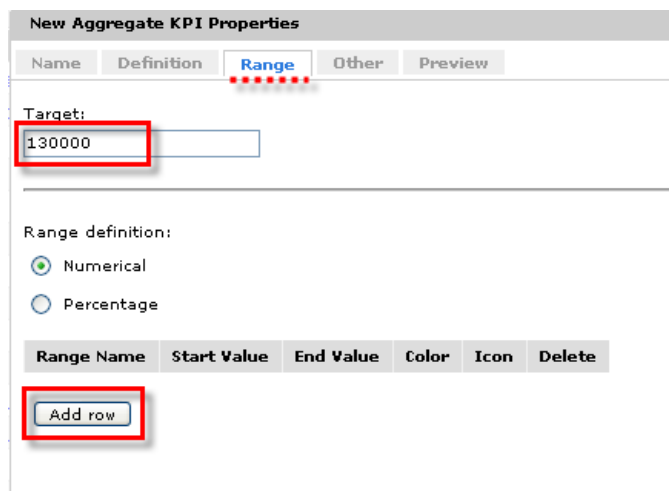
__10.   Click the Range tab.

__a. Enter a target of: 130000

__b. You will add two ranges. Click the **Add row** button to create the first range.

__11. Type the values for the first range:

    __a. Enter a Range Name: **Below Expectation**

    __b. Start value: **80000**   End value: **100000**

    __c. Click the **Add row** button to create the second range.

          (if you prefer to have additional ranges – add as many as you like)

__12. Type the values for the second range:

    __a. Enter a Range Name: **Expected**

    __b. End value: **170000**

                        SOA E2E Workshop - Dashboards

__13. With the range values defined, choose colors for your ranges.

> __a. Click the **Color** icon for the first range.

Range Name	Start Value	End Value	Color	Icon	Delete
Below Expectation	= 80000	< 100000	■	–	🗑
Expected	= 100000	< 170000	■	–	🗑

> __b. Choose a color from the palette, click OK.
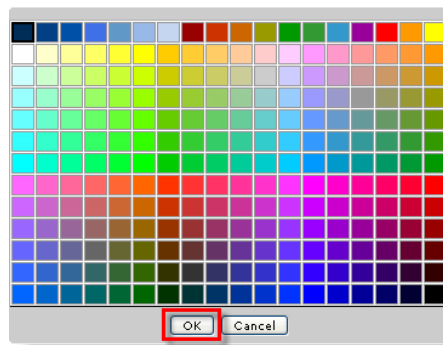
> __c. Repeat, selecting a color for the second range.

__14. To set a status icon for the range, click the **'-'** in the **Icon** column for the first range. The icon you choose for each range will be displayed, depending on the range of the actual value of the KPI.

Range Name	Start Value	End Value	Color	Icon	Delete
Below Expectation	= 80000	< 100000	■	–	🗑
Expected	= 100000	< 170000	■	–	🗑

__15. Click the icon you wish to display when the KPI is in the **Below Expectation** range, click **OK**. Repeat and choose the icon for the **Expected** range.

__16. Examples of colors and status icons are below. Status icons only appear when a Table View graphic is selected for the KPI. You will use this display option for this KPI.

Range Name	Start Value		End Value		Color	Icon	Delete
Below Expectation	=	80000	<	100000	■	●	🗑
Expected	=	100000	<	170000	■	●	🗑

__17. Since this KPI Amt Requested [Min] reflects a dollar amount, you have the option of selecting the currency display. C

__a. Click the tab labeled **Other**.

__b. Check the **Currency** box

__c. Scroll down and select **USD**.
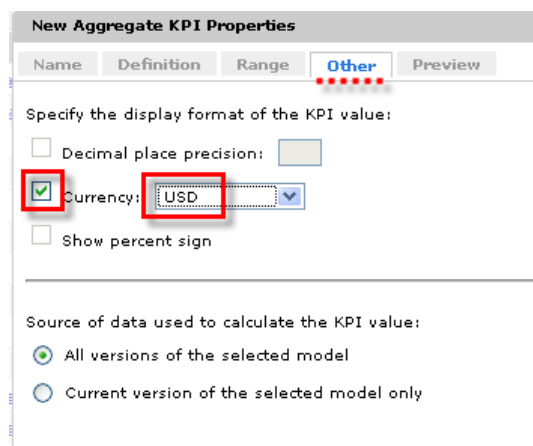
__18.   The Preview tab provides a final look at the new KPI definitions before saving the changes. The default table view graphic is displayed. If everything looks complete, click **OK**.
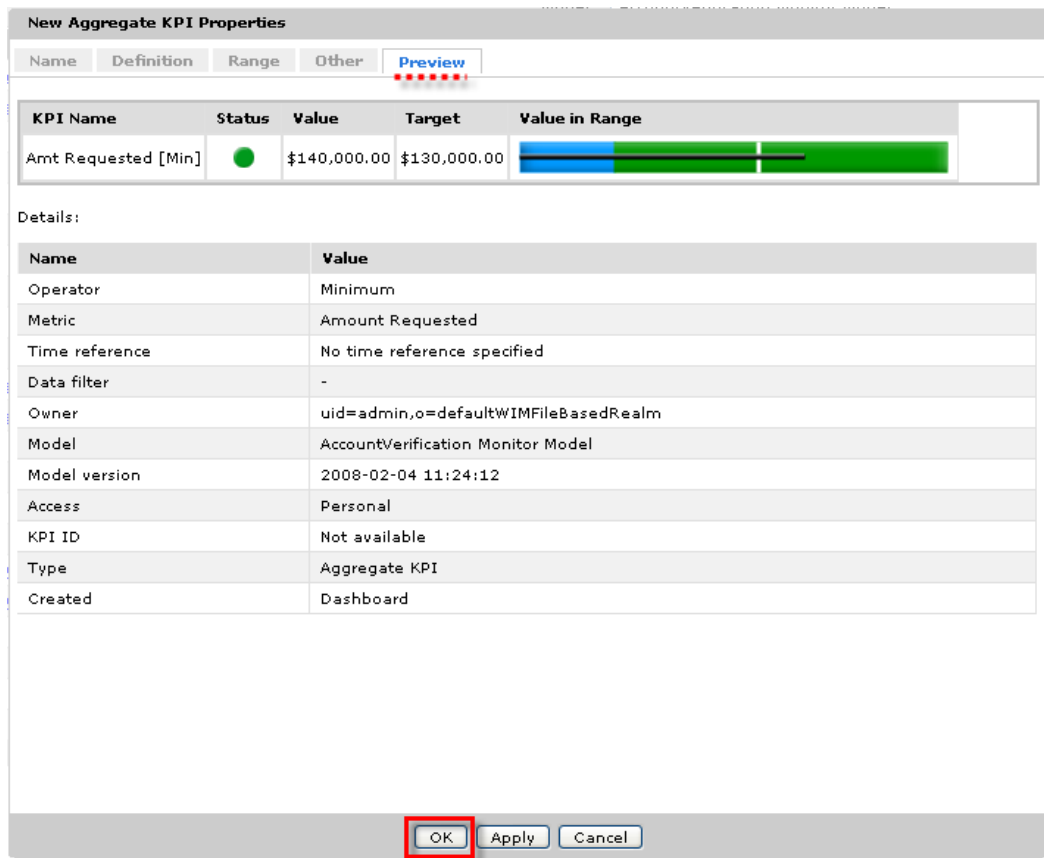
__19.   The KPI properties window disappears, leaving the KPI Manager list. You should now see your new **KPI Amt Requested [Min]** in the list. Notice it is listed as a **Dashboard** created KPI of type **Aggregate**.
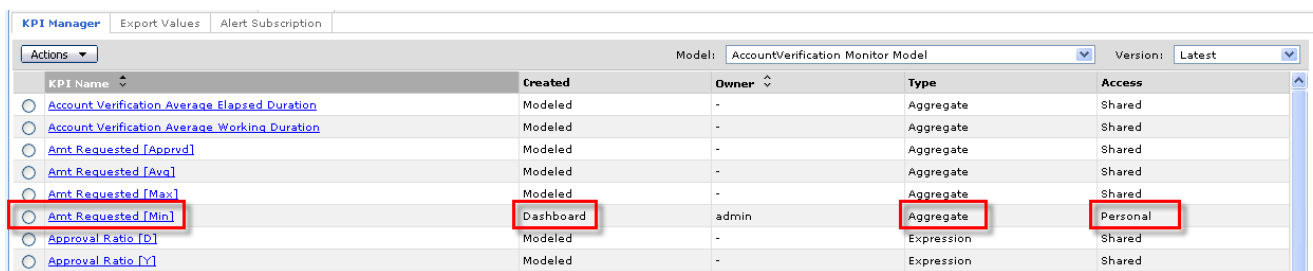
__20. Now you will add this new KPI to your existing **MyDashboard1** page to satisfy the request from the business community. Click back to the **Dashboards** tab and click the **Layout assistance** icon to toggle it on.
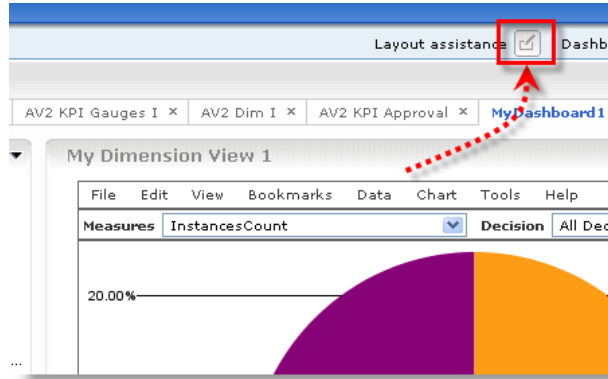
__21. Click the blue link Add to Dashboard below your pie chart. Select **KPIs** and click **OK**.
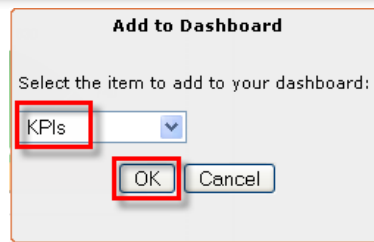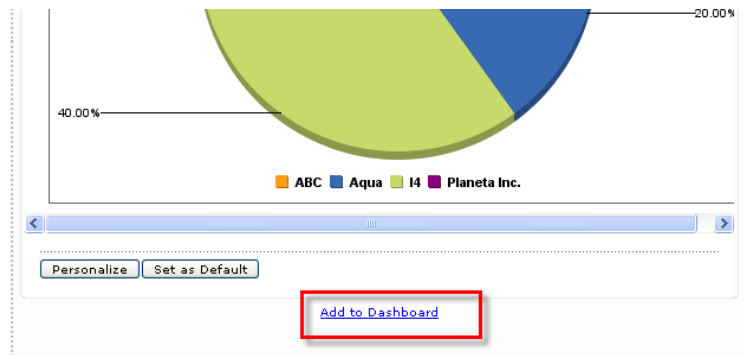
__22. A new empty KPI view appears below the pie chart in the location you specified, click the **Personalize** button to choose the KPI which will populate this view.
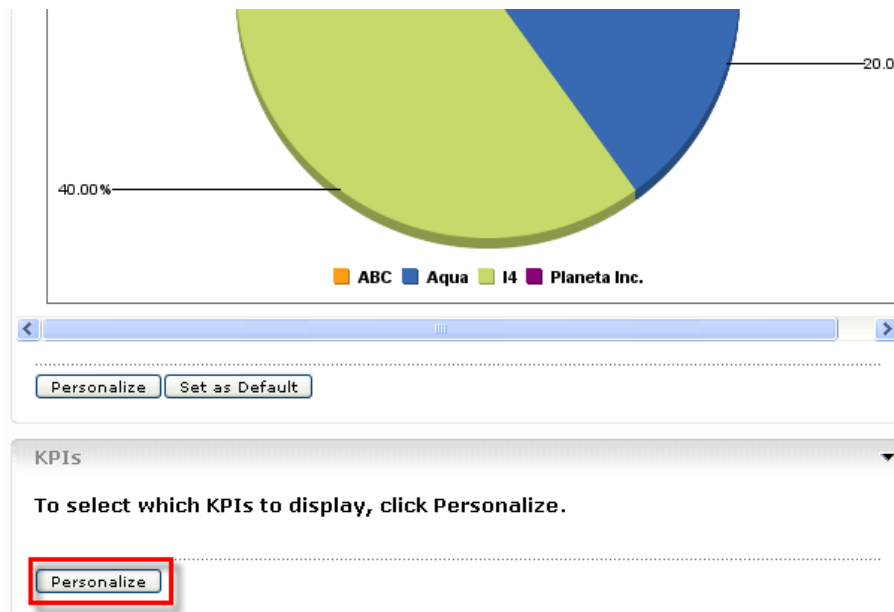
__23. Click the '**+**' to expand the **AccountVerification Monitor Model 2008-02-04 11:24:12**.

    __a. Note: The model may be shown in a different order on your list, make sure you choose the correct model

__b. Scroll down until you find the new KPI you created: **Amt Requested [Min]**
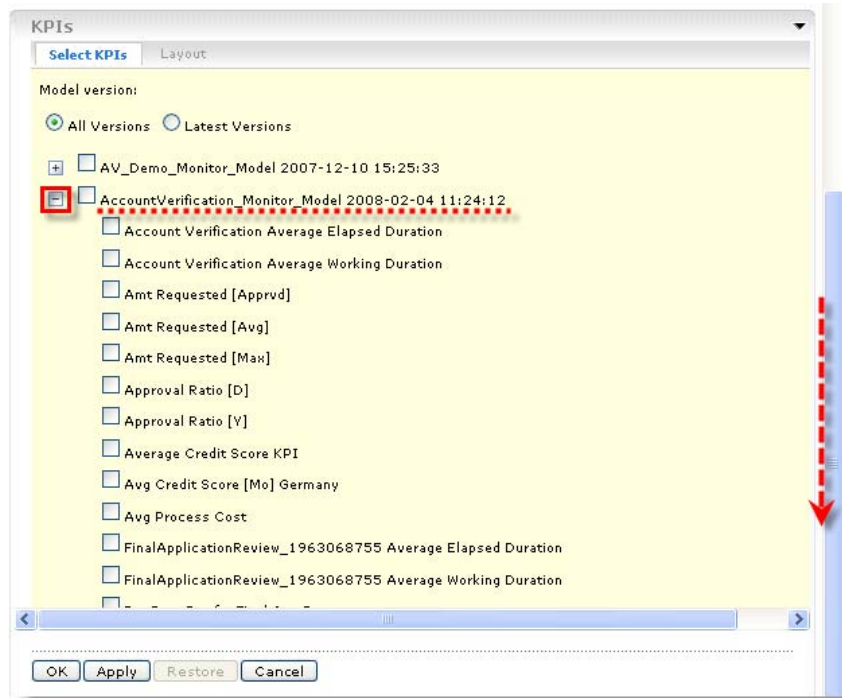
__24.   Place a check next to **Amt Requested [Min]** and click **OK**.
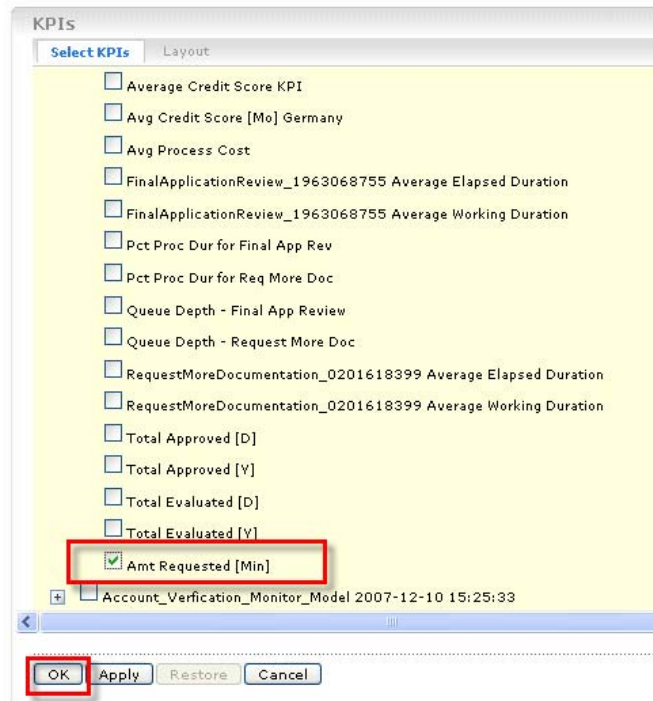
__25.   The default Table View version of the new KPI is displayed. You may try other display options if you wish. Use Personalize > Layout tab > Display mode > Half gauge | Full Gauge | Bar Graph
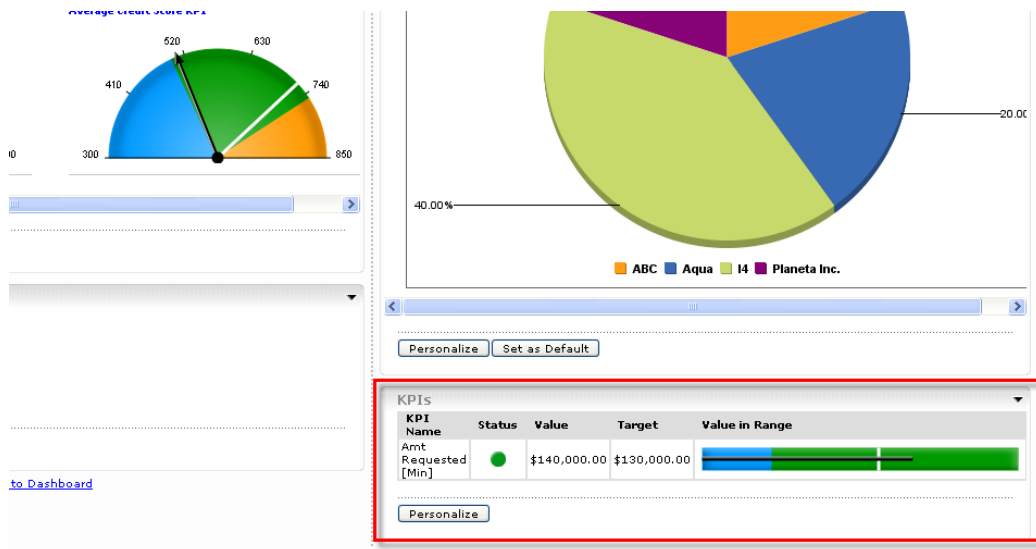
__26.   Notice as soon as your new KPI is displayed, it calculates the value based on the existing process data.

__27.   You may wish to turn Layout Assistance off again to save diagram space, for a cleaner look.
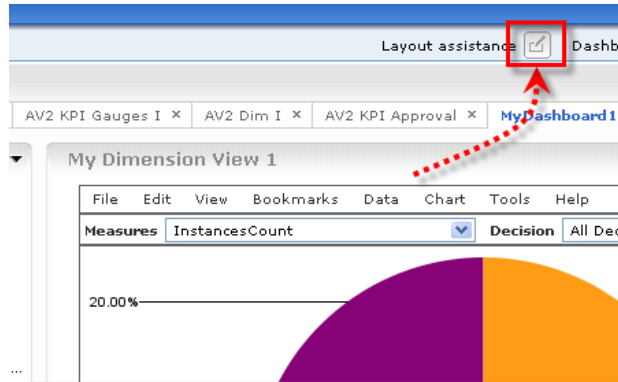
__28. The final appearance of your dashboard, after adding the new KPI should appear similar to the following. Your chart values will be different.
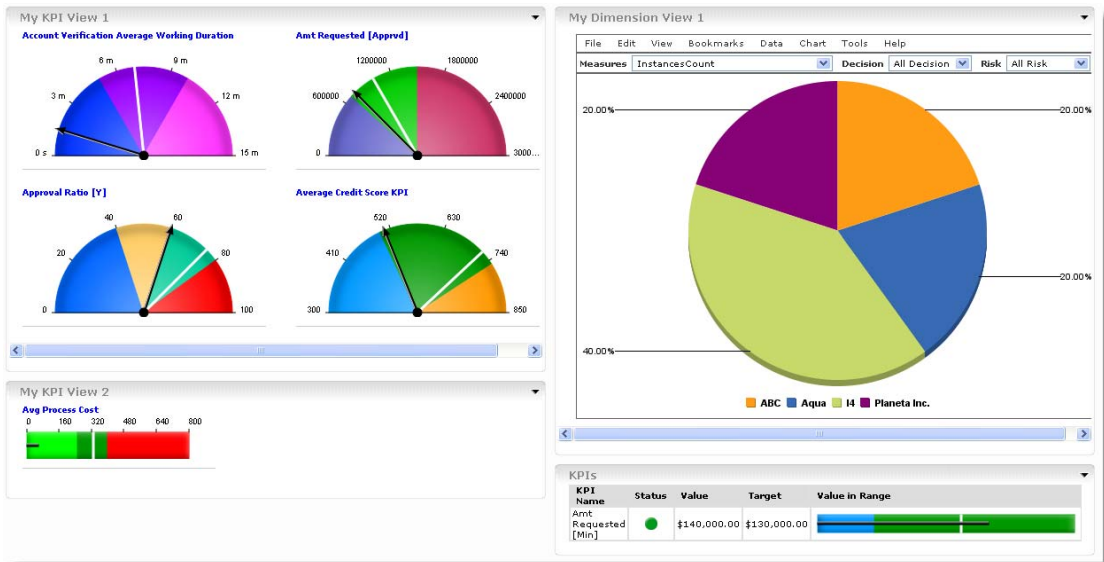
__29. You have completed this section.

Alert Subscription

In this section you will briefly look at the options for subscribing to alerts. WebSphere Business Monitor can be configured to watch for defined Business Situations. These are conditions where some data element being monitored meets a threshold or criteria. Examples of business situations might include:

___xiv. A critical process or series of steps in a process that take too long to complete

___xv. Cost of a process or task, or some other calculated value exceeds or falls short of a threshold

___xvi. Customer data – such as a low credit score or requested product or flagged account information is detected

___xvii. Longer term goals, such as quarter sales don't meet a criteria by a certain date

WebSphere Business Monitor can take action as a result of detecting a Business Situation. One possible action is starting a secondary process that corrects or begins corrective action in response to the situation.

Another possible action is to send an Alert to interested / subscribed parties regarding the business situation that has been detected. Alerts can take multiple forms. Alerts include some basic information about the source of the alerts, then any customized details that were defined in the model.
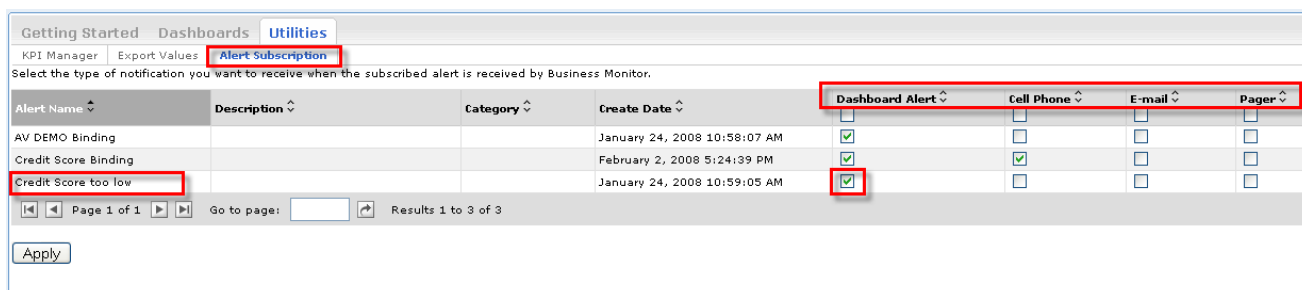
1. Dashboard Alerts – earlier you saw examples of a **Credit Score too low** alert that was sent when the average credit score dropped below a threshold (see page 102)

2. Cell Phone Alerts

3. E-mail alerts

4. Pager alerts

The details to be included in the alert message, along with any customer or process data necessary for the receiving party to take action, are all designed in the Monitor Model.

Users of WebSphere Business Modeler may have specified the requirements for these alert messages, then Integration Specialists created the implementation details using WebSphere Integration Developer as part of the Monitor Model.

This means that when the model was deployed and is running in the system, all the mechanics for creating and sending alert messages is in place. Properly authorized users of the Dashboards need only to subscribe to these messages to receive them.

__1.   Click the **Utilities** tab (next to the Dashboards tab).

__2.   Click the **Alert Subscription** tab to see the available alerts configured in the system.

__3.   Other information regarding Alerts:

- Users may select more than one Alert for which they would receive notifications.

- Users may select more than one notification type for Alerts. (dashboard + cell + e-mail)

- Users may select different notification types for different Alerts. (cell for some, email for others)

- Users may deselect a notification type at any time to unsubscribe.

- Users may subscribe and unsubscribe without I/T involvement.

- Only functional alerts that are defined and running in the system will appear in the list.

- To create completely new alerts available for subscription requires an update to the model, performed by I/T.

__4.   You have completed this section.

Export actuals

In this section you will see the option to export the collected results from monitoring for use as new input to the process modeling environment.

When WebSphere Business Modeler is used to model business processes, the processing time for tasks, the cost of executing the tasks and the probabilities associated with the decisions in the process are used as input to the process simulations. This information is usually based on the assumptions and business knowledge of those most familiar with the operational processes.

When the performance simulations are complete, the output includes summarized and detailed cost and revenue, information on resource usage, information on process paths and other data useful for process design and analysis. This data is used iteratively as the process is refined, new simulations are run; the process is implemented, and now monitored.

As WebSphere Business Monitor runs and collects real information on the execution of the process, it is desirable to feed these actual results back into the modeling process to both validate the original assumptions made, as well as to investigate other potential improvements.

Information like the actual task processing time, probabilities associated with the decisions in the process and potentially cost information can be collected by WebSphere Business Monitor and transferred to WebSphere Business Modeler, where it can serve as input to new simulations for a process model.

The Export Values capability allows the user to choose the data to export from among the various models that are running in the WebSphere Business Monitor system. Different models (perhaps for different processes) as well as different versions of the same model can be chosen to provide feedback data to WebSphere Business Modeler.

The other choice offered is to filter the data such that only the actual results from within a specific time window are exported. This could be useful when seasonal or other fluctuations in processing occur, and it would be helpful to have different known segments of time to use as input to the process modeling environment.

It is likely that only a few users would utilize this capability, then the results might be distributed and used in a structured way in concert with a controlled strategy for process improvement.

__1.   Click the **Export Values** tab under the **Utilities** tab.

__2.   Select the **AV Demo Monitor Model 2007-12-10**…..

__3.   Click the radio button to **specify a time range**.

   __a. From: January 1, 2008

   __b. To: April 01, 2008

__4. Click **Preview Data**.

    __a. Since this file is intended for consumption by WebSphere Business Modeler, it is not formatted for light reading.

    __b. Scroll to the far right of the preview data window, and you will notice a value column. These are the actual values that can be transferred into WebSphere Business Modeler

        __i. Processing Time – durations of tasks, the values you see in the file will be decimal equivalents, converted to time values when imported into the process model.

        __ii. outputSetProbability – the actual decision probabilities that occurred (these will be in pairs for a decision with two branches, triples for three way branches etc.) In the example you see, one branch of the decision was used 14.6%, the other 85.3%.

        __iii. executionCost – a value placed by WebSphere Business Monitor into this field, to be returned and used as the cost for simulation purposes. This information is not automatically determined by the system of course, it must be supplied to the monitor model from an external source or calculated.

__5.   The final step would be to use the Export button to create the file used as input to WebSphere
        Business Modeler. We are not performing the export in this lab.

__6.   Click **Logout** at the upper right of the WebSphere Business Monitor panel.
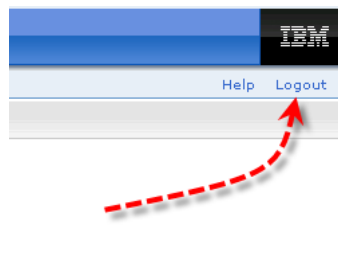
__7.   Close the Firefox browser.

__8.   Double click the **Stop Server** icon on the left side of the desktop. A black window will emerge.
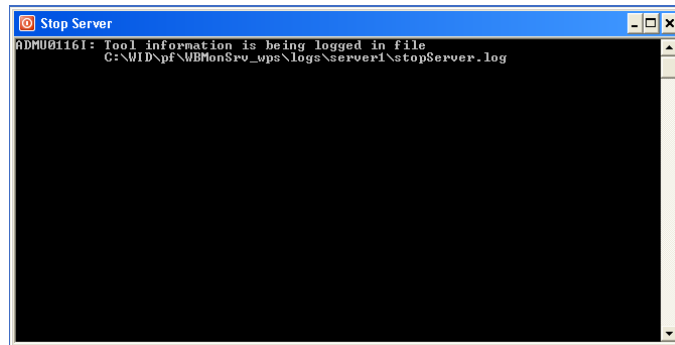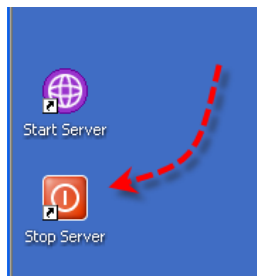
__9.   When prompted by this panel, type **admin** for the User Identity and **admin** for the user password, click **OK**. (this panel will time out if the entry is not made, if this happens, go back to step 8)
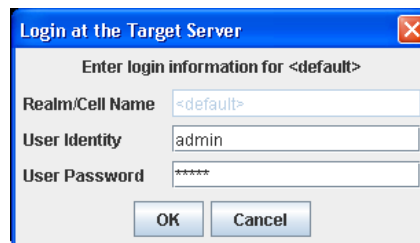
__10.   When the black panel disappears, the server is completely stopped.

__11.   Now that you have seen first hand some of the capabilities of WebSphere Business Monitor, you have the option of continuing with two additional labs:

> Lab 5 Create Business Measures (for Business and Technical users) examines the steps used in WebSphere Business Modeler to prepare a process model with preliminary Business Measures such as those you have seen and used in the Dashboards.
>
> Lab 6 Implement Business Measures (for Technical users) provides an overview of the steps required to create the completed version of the Monitor Model required to provide Dashboards such as those you have used earlier.

__12.   You have completed this lab.

Create Business Measures (Optional)

Do this lab if you have time.

ROLE: Business Analyst

TOOL: WebSphere Business Modeler

In this lab, the existing process model will be reviewed briefly. Some of the anticipated process measurements needed by the business will be introduced. As the Business Analyst, you will then create a subset of the Business Measures needed to provide this information. In a later lab, I/T can reference these suggested measurements as they create an implementation.

Business Measures created in WebSphere Business Modeler are not complete. They provide the implementer an outline of the business measures required and other basic information about potential target values or display options. This lab shows this first phase of defining the business measures, performed by the Business Analyst role. Business Measures must be implemented (completed) in order to be available in WebSphere Business Monitor. The next lab will examine the implementation phase typically performed by an Integration Specialist or similar I/T role.

Using WebSphere Business Modeler, the Business Analyst specifies **WHAT** aspects of the process are to be measured. Later, the Integration Specialist will implement the measurements, specifying the details of **HOW** the measurements are to be calculated.

After the measurements are deployed, and are active in WebSphere Business Monitor, the Business Analyst (properly authorized) can make many adjustments to the measurements, including:

> ___5.   change names, colors, ranges and targets of KPI's

> ___6.   display graphics such as gauges, bars, tables and their colors

> ___7.   create new dashboards and add different views to those dashboards

> ___8.   add new KPI's, based on existing metrics/data

> ___9.   add new KPI's, based on expressions that use existing KPI's

> ___10.   subscribe to alerts

You have seen many of these functions in other labs.

The following major steps are included in this lab:

> 1.   Examine the existing process model
>
> 2.   Plan the Business Measures
>
> 3.   Create the Business Measures
>
> 4.   Review the Monitored Values
>
> 5.   Prepare the model for completion and export for assembly

Examine the process model

In this section, you will review the existing business process model as it has been supplied in a WebSphere Business Modeler project.

__1. Start WebSphere Business Modeler. Double click the **Modeler – Bus Measures** icon on the desktop.

__2. WebSphere Business Modeler will open the Account Verification Project. The AccountVerificationToBe process will be displayed in the process diagram view.
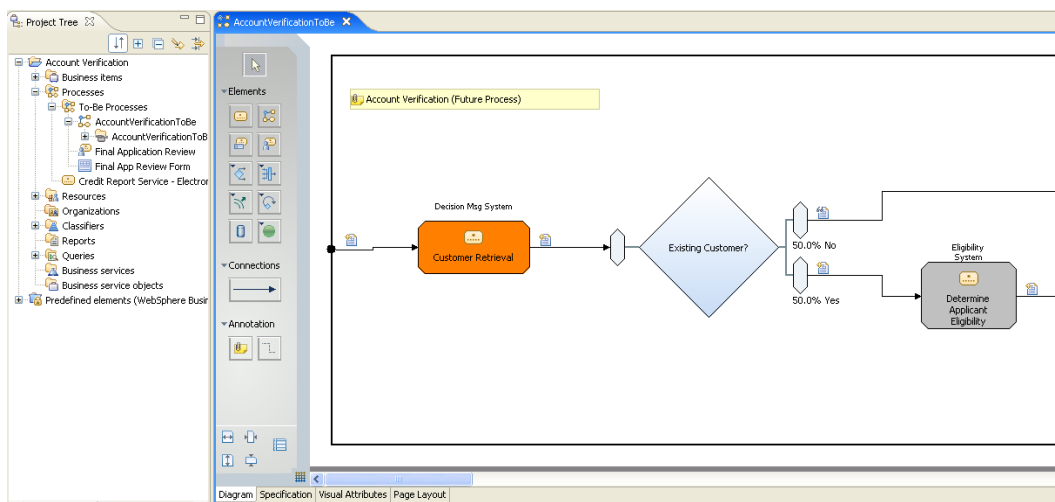
__3. Use the scroll bar and examine the business process. You may also use the 2-pane layout if you prefer.
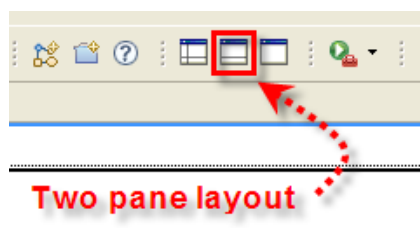
__a. Data arrives where the Customer Retrieval task gathers any prior customer information.

__b. If the customer already exists, the Determine Applicant Eligibility task runs to determine whether a credit report is needed.

__c. A credit report is required for requests from those who are not existing customers, and also required for some portion of existing customers.

__d. Based on the credit report results, a business rule determines the credit risk of the application (High, Medium, Low)

__e. High risk applications require additional documentation. This task, Request More Documentation is a Human Task (requiring a person to perform it).

__f. High and Medium risk applications require a final application review before a final decision is made to approve or decline. For applications not approved, a decline notice is generated. (for declined applications, the process ends here) The Final Application Review task is also a Human Task.

__g. Pre-existing customers not requiring a credit report, and low risk customers are approved without the final application review.

__h. All approved applications utilized a data cleansing service to confirm good customer application details.

__i. Approved applications are routed to the Provide Pricing task after which the acceptance notice is generated. For approved applications, the process ends here.

Plan the Business Measures

In this section you will see examples of business measures for the process you have just reviewed. In a typical environment, business goals and objectives are the result of decisions made at the enterprise, functional or division levels.

To measure the performance of the business process against these goals and objectives, quantitative measurements (business measures) must be defined. These measurements may reflect the general health of the business processes, or could be more specialized to validate the results of new initiatives, changes to business processes or focus on suspected trouble spots. Other measurements may look at the performance of internal or external human resources or even services that are important to business goals.

Once these quantitative measurements are defined, implemented and deployed, WebSphere Business Monitor can report on them, updating them continuously via dashboard views.

Different types of measurements are available when preparing to use WebSphere Business Monitor.

Key Performance Indicator (KPI)

- A high level quantitative measurement used to track a value significant to the business.
- KPI's have a target value based on business goals
- KPI's have range(s) to help validate the measurement against expectations.
- KPI's are based on multiple runs (instances) of the business process.
- Example: Average Process Cost

Metrics (Instance Metrics)

□ A numeric value tracked for a single process instance.

□ Examples: Process Cost, Duration, Amount of Credit Requested,

□ An instance metric might capture for example how long it took to complete order number 101 on August 15 for ABC Company.

□ The values from Instance metrics can simply be displayed in a list for each occurrence of a process or combined (Averaged, Summed, Counted etc) to be used in a KPI, such as Average Process Cost.

Aggregate Metrics

□ Are based on Instance Metrics, and include a calculation (Average, Sum, Count, Min/Max) across many instances of the process.

□ Similar to KPI's, except they are not as visible or important to the business.

□ Do not have targets or ranges

□ Are not displayable in standalone gauges, bar graphs, tables on the dashboards

□ Aggregate Metrics are available in the Dimensional and Report views in the dashboards

The following diagram shows the relationship between Instance Metrics, Aggregate Metrics and KPI's.
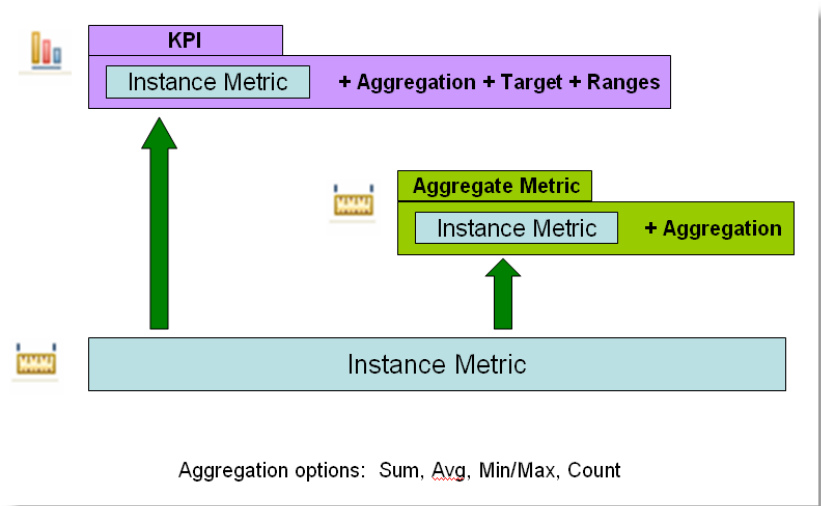
> **Dimensions**
> - □ A non-numeric attribute of data available in a process.
> - □ Examples: Color, Company Name, Location, Product Name
> - □ Dimensions are categories that will be used to help group the data we capture from our processes, so that understanding or examining the data will be clearer.
> - □ Using dimensions, we could look at orders "by Company", or "by Status", or "by Location".

__1.   Several Key Performance Indicators (KPI's) will be used to help measure the performance of the example business process. You will define a small subset of these using WebSphere Business Modeler. In earlier labs you have see these active KPI's in the live dashboards. Review them briefly.

□ Account Verification Average Elapsed duration

□ Account Verification Average Working duration

□ Amt Requested [Apprvd]

□ Amt Requested [Avg]

□ Amt Requested [Max]

□ Approval Ratio [D]

□ Approval Ratio [Y]

□ Average Credit Score KPI

□ Avg Credit Score [Mo] Germany

□ Avg Process Cost

□ Final Application Review – Avg Elapsed Duration

□ Final Application Review – Avg Working Duration

□ Pct Proc for Final App Rev (percentage of the process time spent on the Final App Review task)

□ Pct Proc for Req More Doc (percentage of the process time spent on the Req More Doc task)

□ Queue Depth – Final App Rev

□ Queue Depth – Request More Doc

☐ Request More Documentation – Avg Elapsed Duration

☐ Request More Documentation – Avg Working Duration

☐ Total Approved [D]

☐ Total Approved [Y]

☐ Total Evaluated [D]

☐ Total Evaluated [Y]

__2.   Instance metrics represent information from a single occurrence of the process. In our example, this means a single customer request. There are more instance metrics used in the running process than shown here, but a subset will give you a sense of how they are used. Review the information below, listing some of the instance metrics you have seen in labs. Many KPI's are derived by evaluating these metrics across many process instances.

    ☐   Account Verification Start Time / End Time

    ☐   Account Verification State

    ☐   Amount Requested

    ☐   Company Name

    ☐   Cost – Process

    ☐   Cost – Customer Retrieval (each individual task has a cost metric)

    ☐   Credit Risk (LOW, MED, HIGH)

    ☐   Credit Score (300-850)

    ☐   Customer City

    ☐   Customer Country

    ☐   Decision Status (Approved, Declined)

    ☐   More....

__3.   Aggregate Metrics are similar to KPI's, based on Instance Metrics and include an aggregation function such as Sum, Max/Min, Count and Average. The difference is really how they are used in the dashboards. They represent data that is interesting for analysis, but not necessarily important enough to be considered a Key Performance Indicator in its own right. These Aggregate Metrics are available in the Dimensional views and Reporting views, where slicing and dicing the data is done, but they are not represented by dedicated gauges and bar graphs like a KPI. Examples are below.

☐ Process Cost [Avg]

☐ Process Cost [Min]

☐ Process Cost [Max]

☐ Amount Requested [Avg]

☐ Amount Requested [Total]

☐ More...

__4. Examining a process using different dimensions can provide insight into relationships and trends that might be difficult to see in other ways. Dimensions are ways to separate the data into groups or categories. You will see the following dimensions in the sample process. As you will notice, Dimensions don't represent "new" data; they just provide different ways to examine data that already exists.

Dimensions sometimes have multiple levels. Our dimension called Location has two levels – Country and City. We will also use some specialized dimensions for the two Human Tasks in the process.

Process Dimensions

☐ by Company (Company name)

☐ by Decision (Approved, Declined)

☐ by Location (Country and City)

☐ by Risk (Low, Medium or High)

Human Task Dimensions

☐ by Assigned Owner (the actual user who is handling the human task in the process)

☐ by Status (Running or Completed)

Dimensional Analysis

The ability to separate and view aspects of the process results into different groups quickly is very useful alone.

The power of dimensional analysis becomes even more evident when multiple dimensions are combined. In our process you will for example be able to view combinations such as: the Average Process Cost for all High Risk requests by Decision Status (Approved or Declined). For Human Tasks, combinations such as: View all tasks with their durations by Owner.

__5. Now that you have reviewed some business measures (KPI's, Metrics and Dimensions) that will be used in the model, you will create some of these using WebSphere Business Modeler.

Create Key Performance Indicators (KPI's)

In this section you will add some KPI's to the existing process model.

__1. Click the white space in the process diagram. This indicates you are defining information related to the entire process, and not about a single object that may have been selected while you explored the flow earlier.

__2. Click the tab labeled **Business Measures,** below the process diagram.
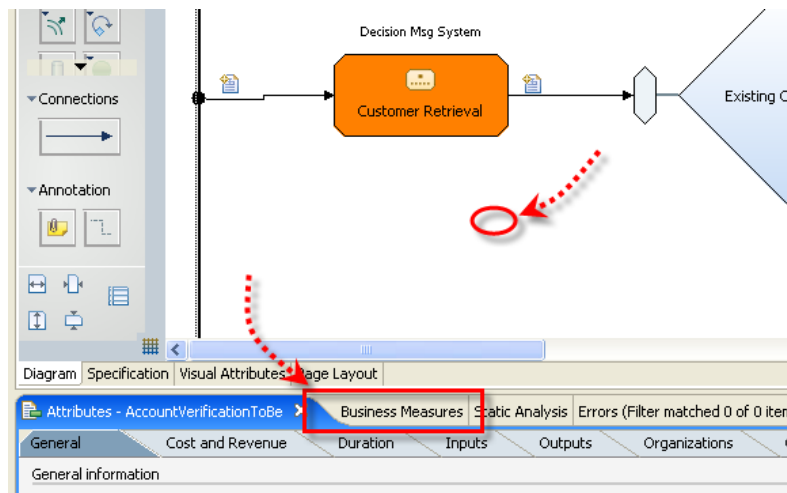
__3. You will now add several KPI's to the model. You should still be on the **Business Measures** tab, and the **Business Performance Indicators** tab should be selected (default). Click **Add**.
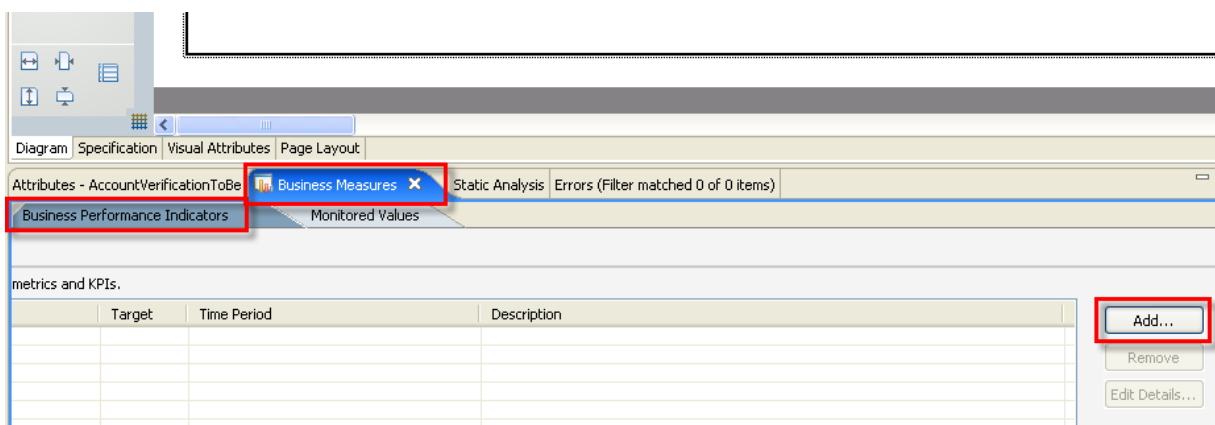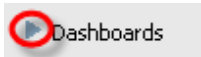
__4.   The Business Measure details panel will emerge with a measure type of Unspecified. If the Business Analyst is unsure of the type of measure needed, then leaving this unspecified is perfectly acceptable. The point is to supply as much information as possible to communicate good requirements to I/T for the implementation of the measurements in the next phase. Notice the ability to use Templates, you will use this later. Select the radio button for **KPI**, and you will see several other options become available.
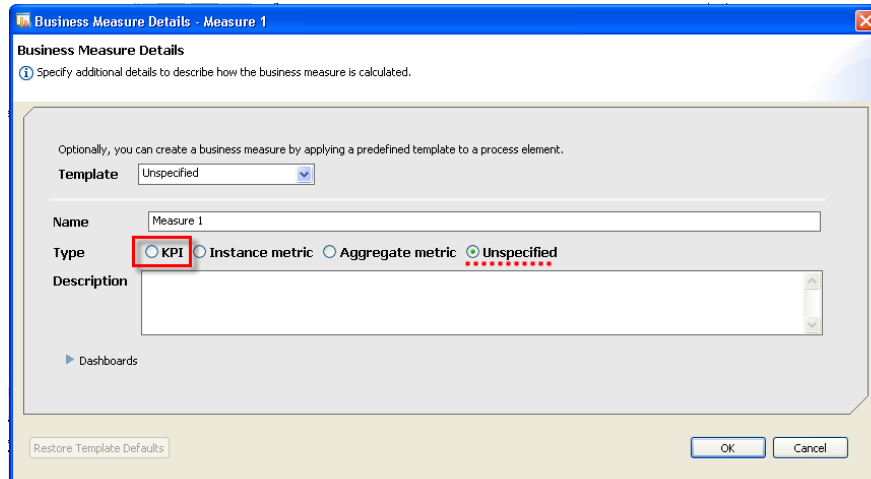
__5.   With **KPI** selected, a scroll bar appears on the right, and you see options to specify a target value as well as details for ranges.
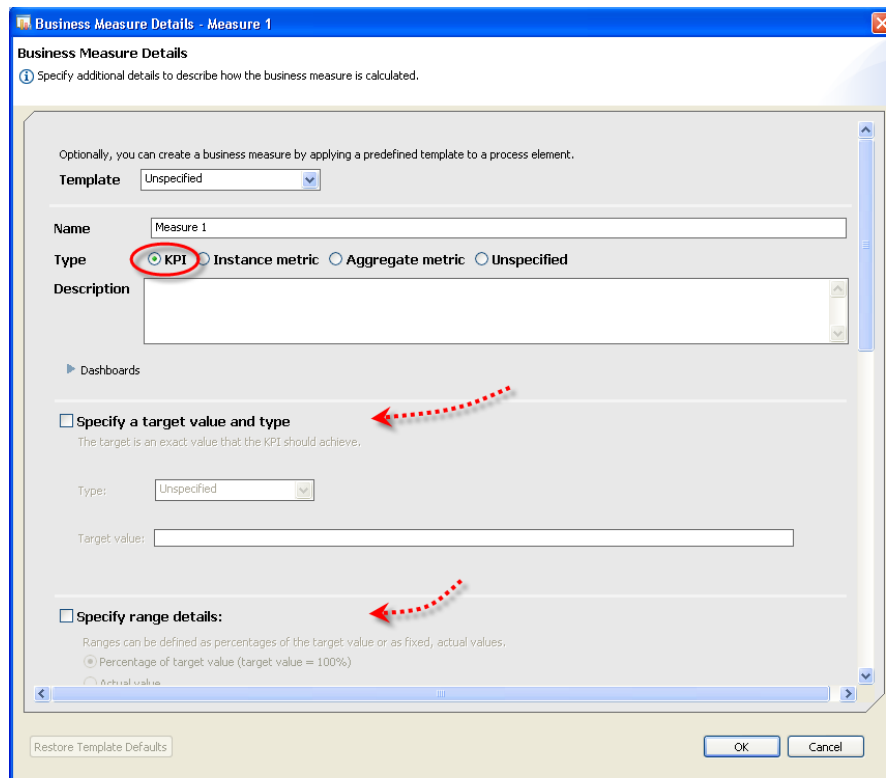
__6. In the **Name** box, enter **Account Verification Average Working Duration**. Enter a description: **Average working duration of the process**. As you begin to create these, remember that the results will be displayed on web dashboards for many business users to reference, so the titles and descriptions should be as clear and concise as possible.

__7. The Business Measure Details panel provides some sample visuals to show how KPI's can be used in different dashboard views. Click the **Dashboards** arrow to expand it. Click the different options. The sample shown is the KPI Half-Gauge. The white line in the gauge displays the Target value; the colored sections represent the Ranges.


__8. Click the dashboard arrow again to collapse the views.

__9.   Specify a target value for this KPI.

__a. Scroll down and click the box to select: **Specify a target value and type**.

__b. Use the drop down arrow to select **Duration**

__c. Use the green up/down arrows to select, or enter **7** in the Minutes box.

__10.   Now specify details for the ranges for this first KPI. (refer to the chart below)

__a. Scroll down and select the box to **Specify range details**:

__b. Select the radio button for **Actual value**

__c. To the right of the range table, click the **Add** button.

__d. Click the cell for Range Name and enter **Low**.

__e. Click the cell for Start value, click the ellipsis box, since it is set to zero, just click **OK**.

__f. Click the cell for End value, click the ellipsis box, select **5 minutes**, click **OK**.

__g. Add two more Ranges for **Med** and **High**, using values from the chart below.
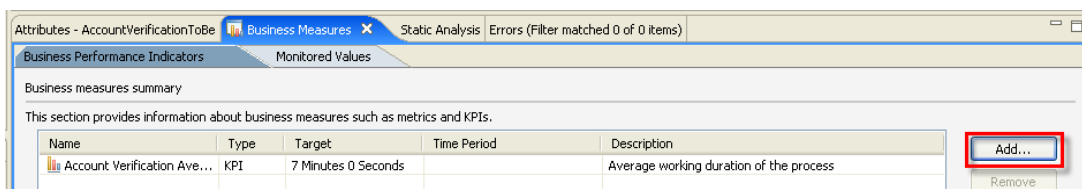
__h. We will not set a time period, alerts or dimensions at this point, so click **OK** at the bottom of the window to complete the definition of this KPI.
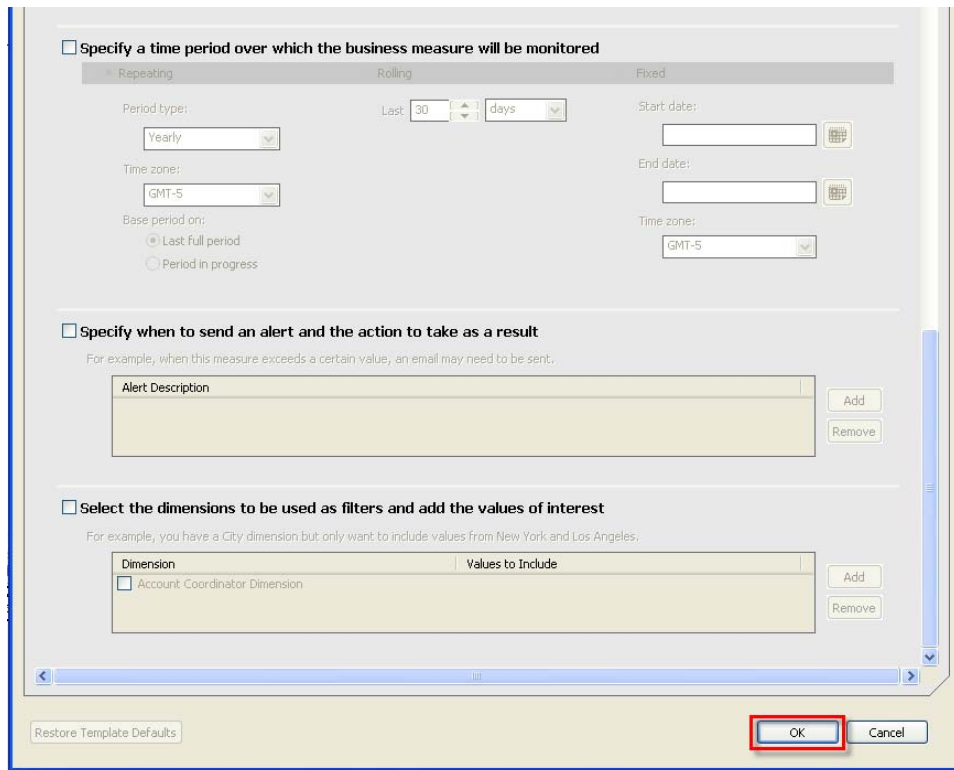
__i. Your new KPI should now be reflected in the Business Measures Summary list. You can click the column boundaries and reposition them if you wish.

__11. Click the **Add** button to add the next KPI.

__12. Repeat the steps beginning with step 4 above, to add a few more KPI's to the model, substituting the information from the chart below. It is not critical to create all of the KPI's, the idea is to get comfortable with the concept of a KPI and the type of information it represents. All of the KPI's shown below will be seen later in a live dashboard view, so it will be helpful if you review and be familiar with them.

> **Note:** The remaining KPI's are Numbers (decimal) rather than Durations, so you will not be prompted with the ellipsis box nor the entry box for hours/mins/secs. You may enter Number values directly in the cells.

KPI Name	Descripti	T	Ranges	Source:
Account Verification Average Working Duration	Average working duration of the process	7 min	Low: 0-5 min Med: 5 – 10 min High: 10 – 15 min	The average of an instance stopwatch metric: *Account Verification Working Duration.*
Amount Requested [Avg]	Average credit amount requested, from incoming customer data	$ 200,000	Level 1: 0 – 300000 Level 2: 300k-500K Level 3: 500K-700K Level 4: 700K-1000K	The average of an instance metric: *Amount Requested*
Approval Ratio [Y]	Percentage of requests that were approved, current year	75	Low: 0-40 Marginal: 40-60 Normal: 60-80 High: 80-100	An expression using 2 other KPI's, *Total Approved* and *Total Evaluated*.
Average Credit Score	Average credit score from the incoming customer data	720	Low: 300-500 Medium: 500-750 High: 750-850	The average of an instance metric: *Credit Score*
Avg Process Cost	Average cost of performing all tasks in	$ 32	Below SLA: 0-250 Meets SLA:	The average of an instance metric: *Cost - Process*

	the process		5	250-400 Exceeds SLA: 400-800	
Queue Depth – Request More Doc	Current number of Request More Doc tasks waiting for handling		3	Excellent: 0-2 Acceptable: 2-4 Marginal: 4-8 Unacceptable: 8-10	The value of an instance metric: *Request More Documentation Queue Counter*
Total Approved [D]	Number of requests approved, current Day		8	Low: 0-5 Med: 5-10 High: 10-25	Value of an instance metric: *Approved*, acting as a counter for approved requests.

    __a. Note the naming convention **KPI Name [??]**. This is sometimes used to denote a time period, or method of calculation such as averaging [Avg], to make the meaning of the KPI more clearly to those reading it on live dashboards. Using this convention is optional.

    __b. Notice the variation in the names of the ranges. They need only have meaning to the intended audience. In many cases, Low, Medium, High might not convey the appropriate meaning, so more descriptive or specific language is used.

    __c. KPI's are derived from different sources. In some cases, they are Averages or Sums or Maximums taken directly from customer data flowing through the process, such as Average Credit Score. In other cases the values are calculated from internal information that has no meaning to a customer, such as the cost of the process (Cost – Process). A third option for creating KPI's allows the definition of an expression that uses other KPI's. In our example the KPI called Approval Ratio is an expression KPI. This level of detail is not necessarily important to the business analyst creating the KPI, but when I/T implements the KPI, this flexibility can be very powerful.

    __d. As the business and/or the processes change, new measurements can be added or deleted, targets and ranges can be changed, and new KPI's can be added dynamically and/or deployed with new versions of the process.

__13. The KPI called **Total Approved [D]** represents the total number of approved requests in a single day. This is a good example where setting the time period is helpful. In the definition of that KPI, if you scroll down and check the box: **Specify a time period over which the business measures will be monitored**, you will see these options. Notice the options also exist for Rolling and Fixed time periods.

__14.   After you have added KPI's, your Business Measures summary should look as below:

Create Instance Metrics

In this section you will create instance metrics, which represent data related to a single instance of the process. In our example, this means a single credit request sent by one company. These metrics can be useful in two ways. First they can be viewed directly on the dashboards in an Instances view where specific data related to a single process can be seen. e.g. application ID, amount requested, company name, credit score etc. Secondly, the instance metrics provide the data from which KPI's or Aggregate Metrics are calculated.

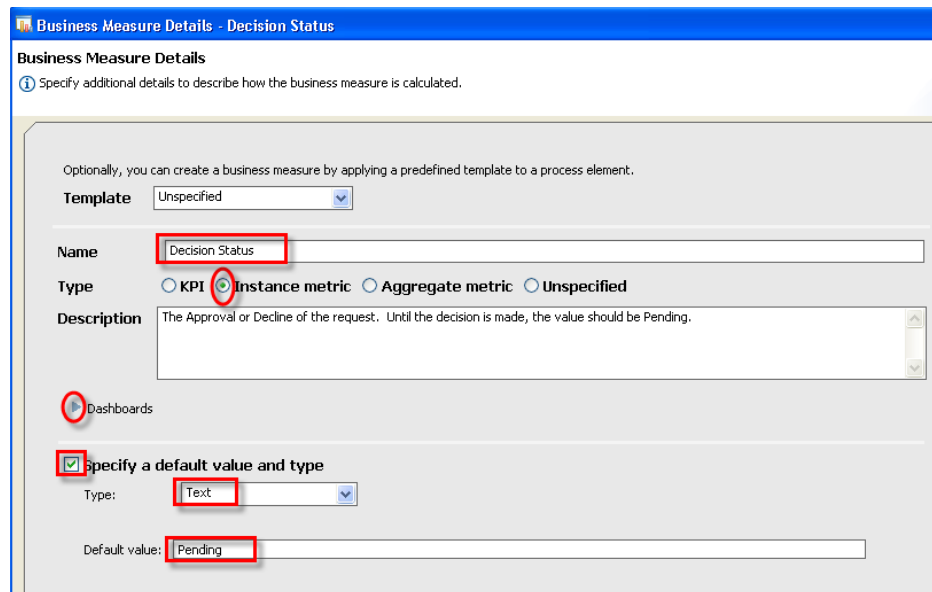__1.   In the Business measures summary, click the **Add** button.

__**2.**   Select the radio button for **Instance metric;**

   __a. Name: **Decision Status**.

   __b. Description: **The Approval or Decline of the request. Until the decision is made, the value should be Pending**

__c. Default value & type: **Text**, **Pending**

__3. Click **OK**. Your results should appear as below, showing your new instance metric.

__4. You have just created a *custom* instance metric. You will now create another instance metric with an alternate method using templates. Templates make it faster to create frequently used metrics. You will create the instance metric for **Company Name**. This metric will get its values from the data coming into the process.

__5. Optional: Data flowing through the process is represented in the process model by **Business Items**. You may examine the **Customer Application** business item before proceeding.

__a. Click the '**+**' to expand the **Business Item** catalog in the project tree and double click on the **Customer Application** business item to view its contents. You may need to switch to the 4 Pane layout to see the project tree.

__b. In the **Customer Application** tab, expand the customer section. Note the **companyName** and **creditScore** fields are already defined here. The new metrics you create will refer to these fields from the business item. Close the tab when you have reviewed this.
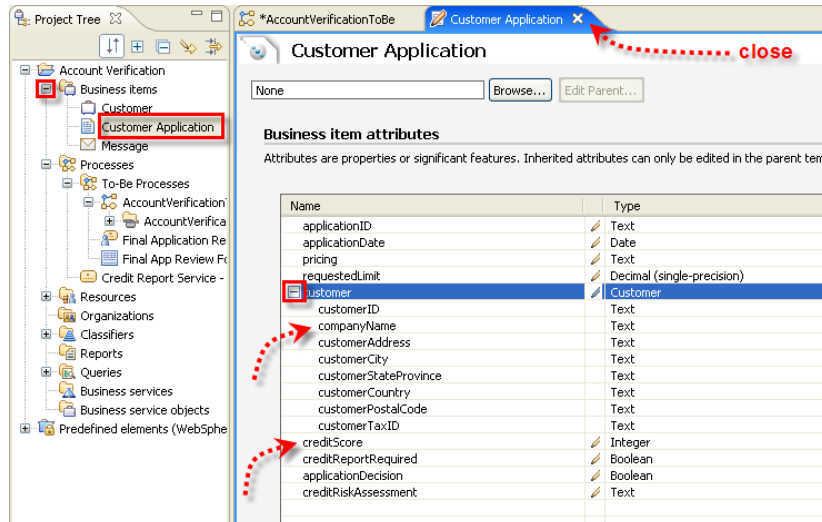
__6.   To create the instance metric for **Company Name**, from the process diagram view:

__a. Right click white space in the process diagram

__b. Select Create **Business Measure > Predefined > Business Item Input**
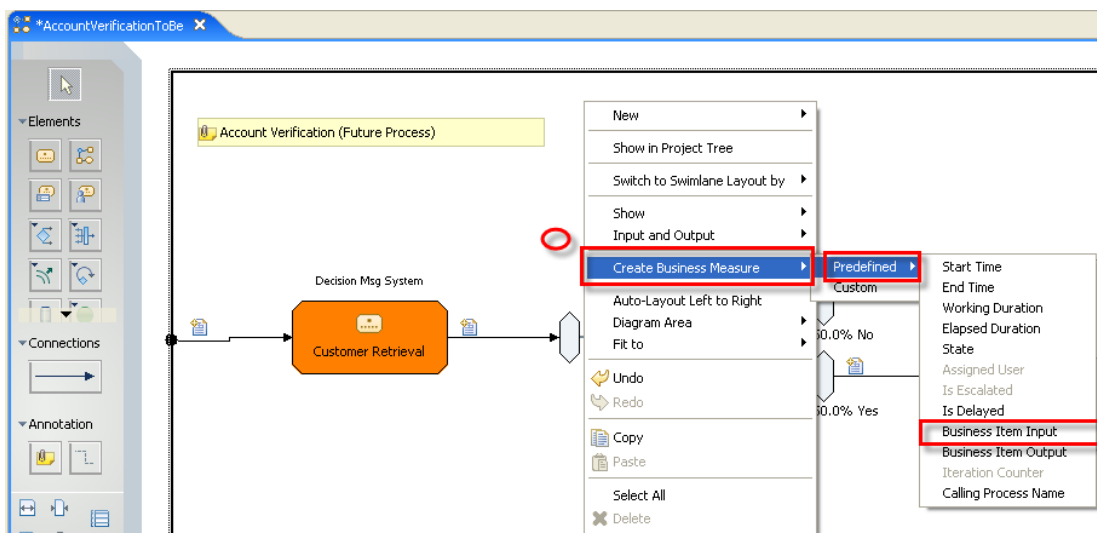
__c. In the Business Measures detail window, note that several fields are already populated for you. The Attribute box shows an error because it is incomplete. To point this metric to the business item mentioned earlier, click the **Browse** button next to the Attribute box.

__d. Select the **companyName** field/attribute and click **OK**.
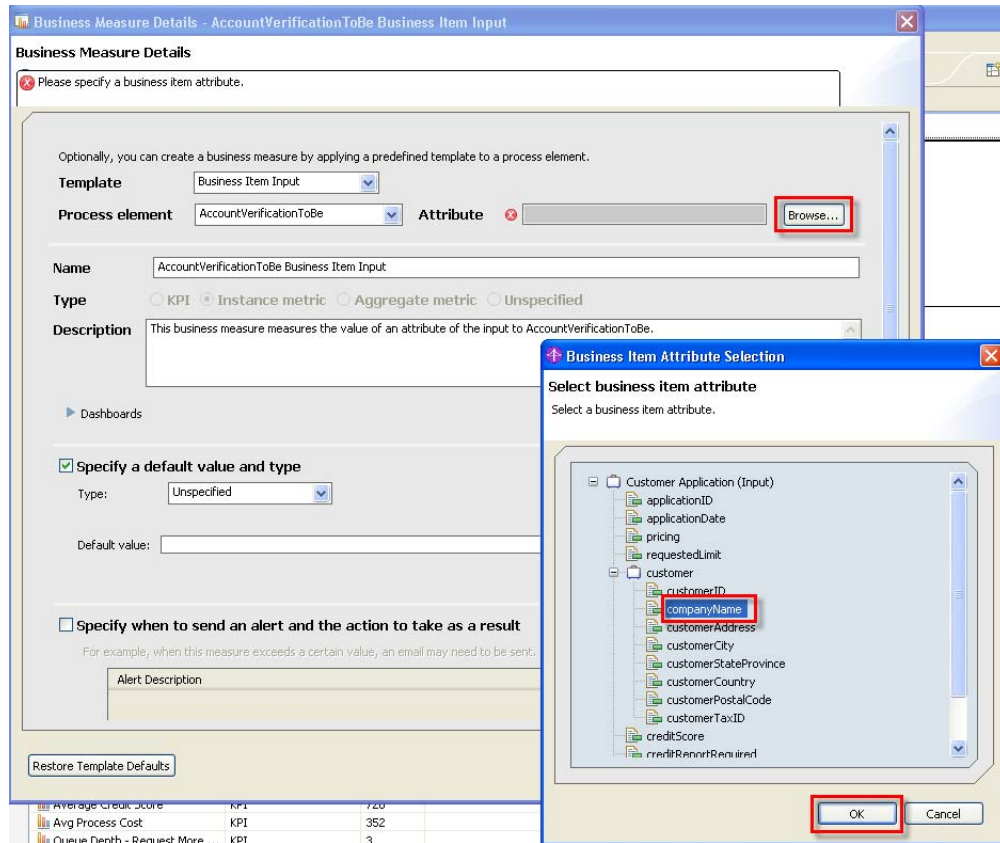
__e. Now this metric refers to the business item companyName, so this data will get picked up from the incoming data when the process starts.

__f. Change the default Name of the metric to **Company Name**, and modify the default description to read "**This metric stores the value of companyName from the incoming business item**".

__g. Note the type has already been set for you.

___h. Click **OK** to complete the definition of this metric. The Business measures summary should appear as below. Note the icon is slightly different to represent instance metrics created from templates.

___7. This chart shows some of the key instance metrics. Several others existed in the models that were running, you will not take the time to create them here.

Instance Metric Name	Description	Type	Source
Decision Status	The Approval or Decline of the request.	Text	This value is added to the captured data when the Generate Pricing & Approval or the Generate Decline tasks run in the process.

Company Name	Name of the company making the request.	Text	Actual data from the incoming customer request.
Credit Score	The score returned for this specific company.	Decimal	Actual data from the incoming customer request or added to the process data later.
Cost - Process	A dollar value representing the cost of an individual process instance.	Decimal	A separate cost is calculated for each task that completes in the process. This metric is the sum of the individual task costs. These costs are based on the time required to perform each task, and since two of the tasks pause and wait for human interaction, the costs will vary each time the process runs.

Create Aggregate Metrics

You will now create an aggregate metric. Aggregate metrics evaluate Instance Metrics across multiple process instances and combine (aggregate) the results using a calculation (aggregation function) such as Average, Sum, Min, Max or Count.

Aggregate metrics are similar to KPI's in the way they are calculated, but are reflected differently in the dashboards. KPI's can be thought of as specialized aggregate metrics that have a target, ranges and distinct graphical representations on the dashboards (gauges, bar graphs), because of their importance to the business.

__1.   To create an Aggregate metric:

    __a. From the Business Measures summary, click the **Add** button.

    __b. Click the radio button for **Aggregate metric**, the panel changes to offer the correct options

    __c. Enter the metric Name: **Amount Requested [Total]**

    __d. Enter a description: **Sum of the Amount Requested instance metric**

    __e. Optional: Click the dashboards arrow to examine how aggregate metrics might appear on a dashboard.

    __f. Click the check box for **Specify how this measure is aggregated across multiple runs of the process**, and select **Sum** as the Function.

__g. Click **OK** to complete this Aggregate metric. The Business measures summary is updated to reflect this new entry.

__2. Several other aggregate metrics are used in the example dashboards; you will not take the time to create them. Here are some of the key Aggregate metrics.

Aggregate Metric Name	Function	Description
Amount Requested [Total]	Sum	The sum of the values in the Amount Requested instance metric, across process instances.
Credit Score [Avg]	Average	The average of the values in the Credit Score instance metric, across process instances.

Process Cost [Min]	Minimum	The minimum value for the Process Cost instance metric found across process instances.

Create Dimensions

WebSphere Business Monitor captures data which can be analyzed in different dimensions. For example, your new KPI for the duration of the process could be examined further by breaking down the KPI results by the Location of the customers represented. Other examples are to examine the process duration by the Amount Requested or by the Credit Risk of the customer.

You will now indicate which dimension(s) are important for analyzing the measures you have created

__1.	In the Business measures summary, click the first KPI, **Account Verification Average Working Duration** to select it. Click **Edit Details**.

__2.	Scroll down and check the box next to: **Select the dimensions to be used as filters and add the values of interest.** Click the **Add** button**.**

__3. Type over the highlighted default name and enter **Company** for the first dimension.
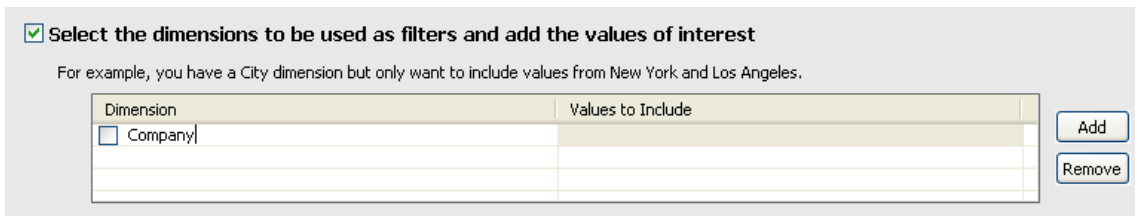
__4. Repeat the **Add** to create **Decision, Location** and **Risk** dimensions.
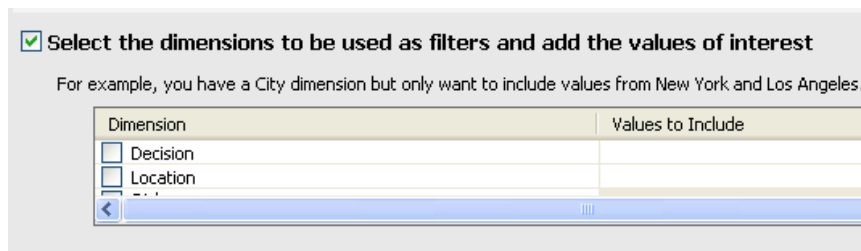
__5. Check the boxes for all dimensions you have added, to indicate they are used with this KPI.
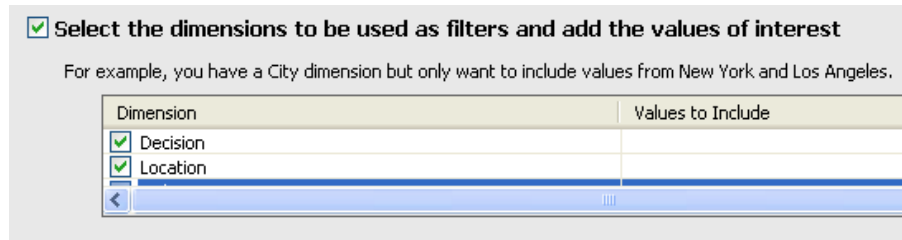
__6.   You will not include any filtering values for now. Click **OK** to complete.

__7.   Back in the Business measures summary view, click the next KPI, **Amount Requested [Avg]** to select it.

> __a. Click **Edit Details**.

> __b. Scroll down and examine the dimensions section at the bottom. The dimensions you added earlier are now available to this KPI as well. You need not make any changes here. The point is to understand that Dimensions are available for analysis across any KPI's and Aggregate metrics once they are defined.
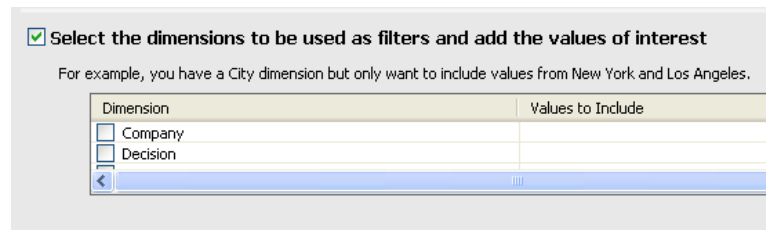
> __c. Click **Cancel**.

__8.   Some dimensions also apply to the specialized human tasks in our example business process. The charts below show the Dimensions both for the entire process and the human tasks that you will examine later in the dashboards.

Process Dimensions:

Dimension Name	Definition	Purpose
Company	Name of the company making the request.	To separate the process data by Company.
Decision	Consists of "Approved" or "Declined"	To separate the process data into two groups, "Approved" and "Declined"
Location	Level 1 – Country	To separate the process data into groups by Country, and then by City if needed.

	Risk	Requests are rated with LOW, MEDIUM or	To separate the process data into groups according to their risk categories.

Human Task Dimensions:

Dimension Name	Definition	Purpose
Assigned Owner	Name of the user assigned to the specific task	To separate tasks by Owner.
Status	Running or Complete	To separate the human task data into two groups by Status.

Create Alerts

In this section you will create an alert based on one of your existing KPI's. In this scenario, the business has indicated they would like to be notified when the **Average Credit Score** KPI drops below 500. To add this alert, you will open the details of the existing KPI, and add the alert details to the proper section.

__1.   In the Business measures summary, click the line for the **Average Credit Score** KPI. Click the **Edit Details** button.

| Name	Type	Target	Time Period	Description
Account Verification Average Working Duration	KPI	7 Minut...		Average working duration of the process
Amount Requested [Avg]	KPI	200,000		Average credit amount requested, from incoming customer data
Approval Ratio [Y]	KPI	75		Percentage of requests that were approved, current year
Average Credit Score	KPI	720		Average credit score from incoming customer data.
Avg Process Cost	KPI	352		Average cost of performing all tasks in the process.
Queue Depth - Request More Doc	KPI	3		Current number of Request More Doc Tasks waiting for handling

Business measures summary — This section provides information about business measures such as metrics and KPIs. Buttons: Add..., Remove, Edit Details...

__2.   Scroll down in the Business Measure Details panel and check the box to **Specify when to send an alert and the action to take as a result**. Click the **Add** button.

☑ **Specify when to send an alert and the action to take as a result**
For example, when this measure exceeds a certain value, an email may need to be sent.

Alert Description

Add
Remove

__**3.**   When a new Alert is added, the description picks up the name of the KPI or other measure which was selected, so the default Alert Description will be Average Credit Score Alert, since this is the KPI you are working in. Add some text this description some text indicating what the alert should do: **Average Credit Score Alert - notify via dashboard alert when this KPI drops below 500**

☑ Specify when to send an alert and the action to take as a result
For example, when this measure exceeds a certain value, an email may need to be sent.

Alert Description
Average Credit Score Alert - notify via dashboard alert when this KPI drops below 500

__4.   Click **OK** at the bottom of the panel. As the Business Analyst, you could supply additional information if you wish, such as who the alert should go to, or specify whether the alert is to be an email alert etc, but as with the other parts of the model, the details of this implementation are supplied by I/T in a later phase.

__5.   WebSphere Business Monitor - Alert notifications can be sent to interested parties via:

       __a. Dashboard Alert view for web based users

       __b. E-mail

       __c. Pager

       __d. Cell

__6.   After the alert has been defined, and is running, authorized users can subscribe to alerts dynamically as needed.

__7.   This ends the work on KPI's, Metrics, Dimensions and Alerts you will do in WebSphere Business Modeler. A couple of points to remember:

       __a. You are not required to enter KPI's, Metrics, etc in WebSphere Business Modeler in order to use WebSphere Business Monitor. The use of Modeler in this regard is optional. This means that all the preliminary definition of KPI's, Metrics etc will occur in the WebSphere Integration Developer Tooling used by I/T, as well as the implementation details.

       __b. While this approach is viable, it does not provide the Business Analyst community the opportunity to supply these requirements in a structured way that is captured with the business process model.

       __c. Obviously, there is a wide range of detail and definition that can be provided in the business measures using WebSphere Business Modeler. Just as with modeling and deploying the process itself, good planning, coordination and communication between Business and Technology will minimize the effort and ensure a higher quality set of measurements that provide tangible, insightful benefits to Business and can integrate well with Technology plans and strategies.

i	**Business Measures created in WebSphere Business Modeler are not yet complete. They require additional implementation information beyond the scope of the business analyst role.**
	
	The business measures information in Modeler does NOT include:
	  ▪ how the KPI's are calculated
	  ▪ the data required to populate the KPI
	  ▪ the views which will be used to display the KPI in a dashboard
	
	These implementation level details are provided by the integration specialist prior to deployment.

__8.   You will look at one additional concept called "Monitored Values", then export the model for the next phase of work by I/T.

Review the Monitored Values

When WebSphere Business Modeler and Monitor are used together, it is possible to calculate and store some specific information from the running process that can be used later for process improvement.

This data is captured by WebSphere Business Monitor, and can be brought back into WebSphere Business Modeler to be used in new simulations, for continuous process improvement.

The specific data chosen for monitoring and feedback is referred to as "Monitored Values". WebSphere Business Modeler allows the user to specify which tasks and decisions in the process should be marked for later monitoring and feedback.

You will briefly review the Monitored Values available for storing in Monitor and feedback into Modeler.

__1.   Within the Business Measures tab, click the **Monitored Values** tab to the right of the Business Performance Indicators tab.
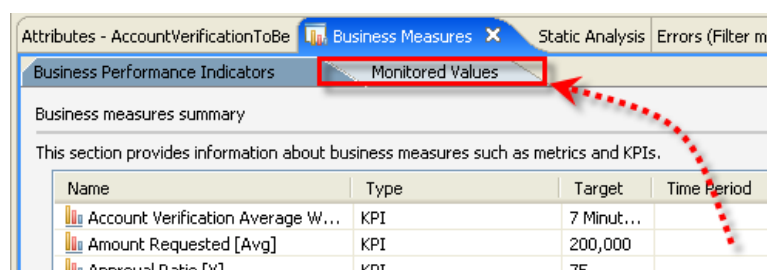
__2.   On this panel, you can see that for the process itself (AccountVerificationToBe), and for each task in the process, it is possible to capture the Processing time and the Processing and Startup costs and any Revenue. For each decision in the model (such as the Approved? decision) the percentage branch information can also be captured for feedback after monitoring.
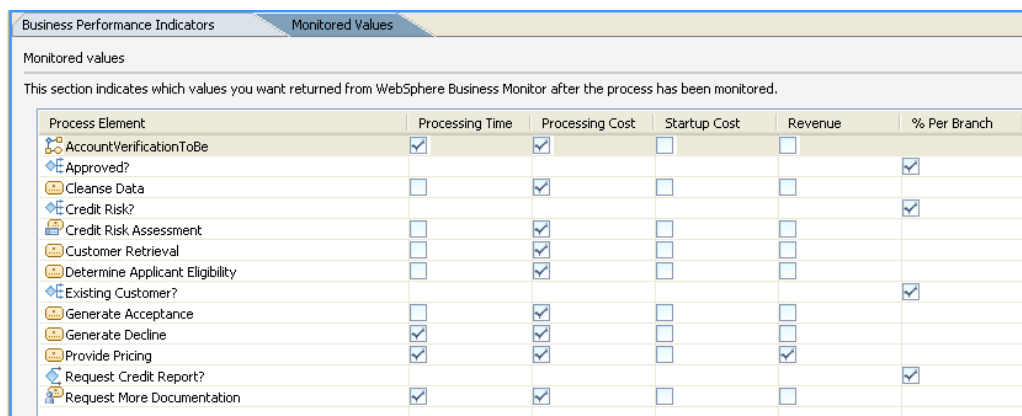
__3.   By checking the boxes for these items, you are specifying the data that you expect to bring back from WebSphere Business Monitor into Modeler for process improvement.

__4.   When the actual Monitor Model is constructed in the next phase, the implementation details must be specified to calculate costs, based on information that is available when the process is running.

__5.   Save your work. Press **CTRL + S**, or click the diskette/**Save** icon on the top toolbar. The asterisk (*) should disappear once you have done this, indicated all changes have been saved.
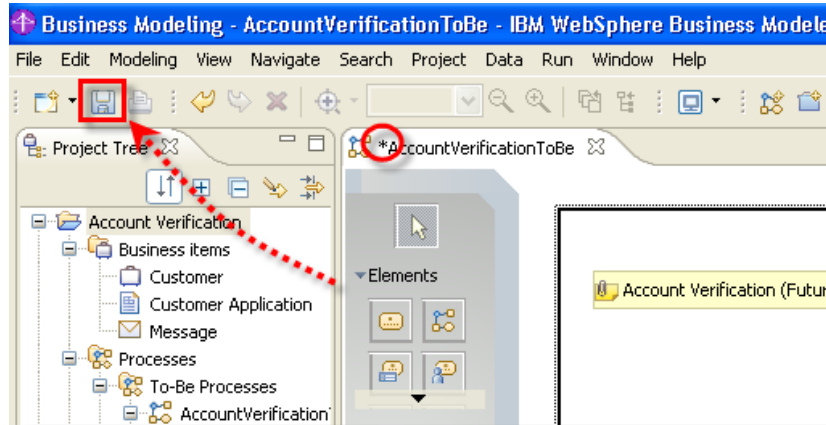
__6.   You have completed this section.

Prepare the model for completion and export

The process model is now complete from the Business Analyst perspective.

WebSphere Business Modeler provides the capability to set some implementation details to further minimize the work in the next phase if desired. These tasks are not typically performed by the Business Analyst, but by the Integration Specialist role.

Since these additional steps are related to the execution of the business process itself, and not to monitoring specifically, we will not perform them in this lab. For the purpose of this lab, we will assume that if it was desirable to make any changes to the implementation details of the process in WebSphere Business Modeler, they have been completed.

The next phase is Assembly. In this phase, Integration Specialists or similar roles perform the following:

___ b.   Complete the assembly of the process itself, specifying the underlying components, services, human task details and business rules that will fulfill the requirements of the business process model. This is performed in WebSphere Integration Developer, using files exported from WebSphere Business Modeler.

___ c.   Generate and complete a Monitor Model that specifies the details of how the business measures will be calculated and displayed. This is performed in the Monitor Model Editor, a plug-in running in WebSphere Integration Developer, using files exported from WebSphere Business Modeler.

You are now ready to use WebSphere Business Modeler to export the files necessary for the next phase.

__1.   Close the process model. Click the **X** on the **AccountVerificationToBe** tab at the upper left of the process diagram.
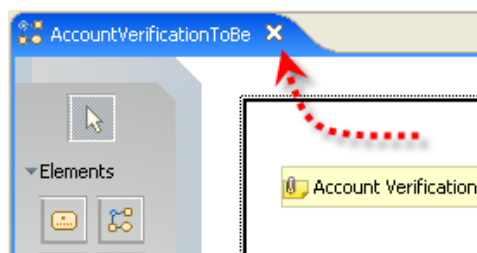
__2.   Right click the **Account Verification** project in the project tree and select **Export**.

2-pane layout vs. 4-pane layout

If you have been using the 2-pane layout, you will need to switch to the 4 pane layout to see the project tree.
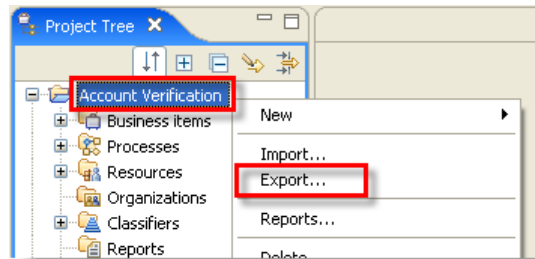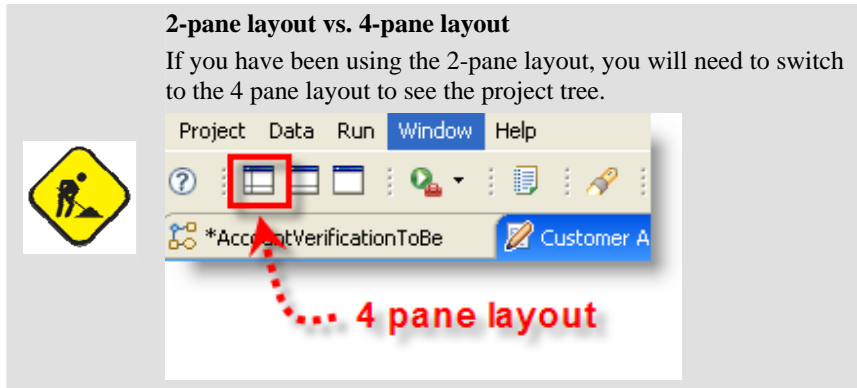
__3.   Select **WebSphere Business Monitor Development Toolkit (.mm)** and click the **Next** button.
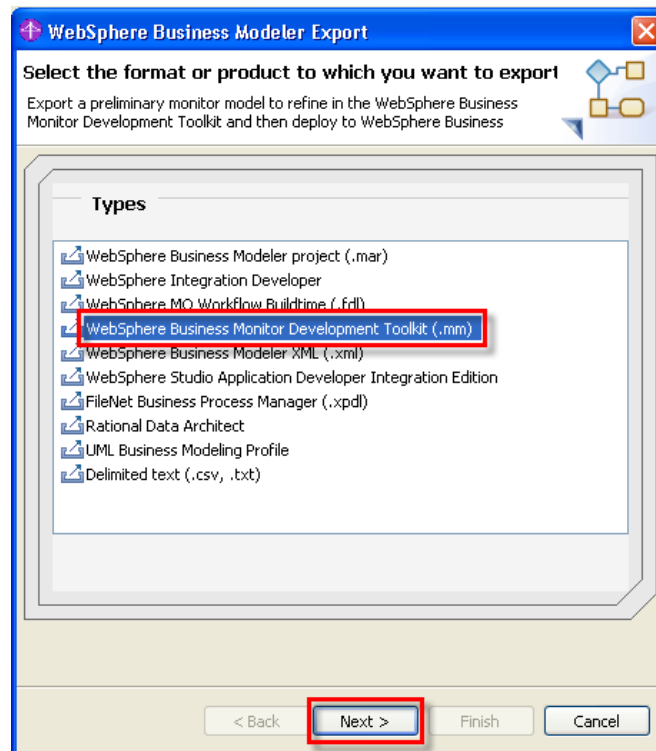
__4.   On the Destination and Source panel,

    __a. Use the drop down arrow to select **C:\Labfiles** as the target directory, or Click the **Browse** button and navigate to C:\Labfiles\ as the target directory.

    __b. Make sure that **Account Verification Project i**s the Project selected.

    __c. Select the button for **Export entire project**.

    __d. Check the box to **Overwrite files**.

    __e. Click the **Finish** button.
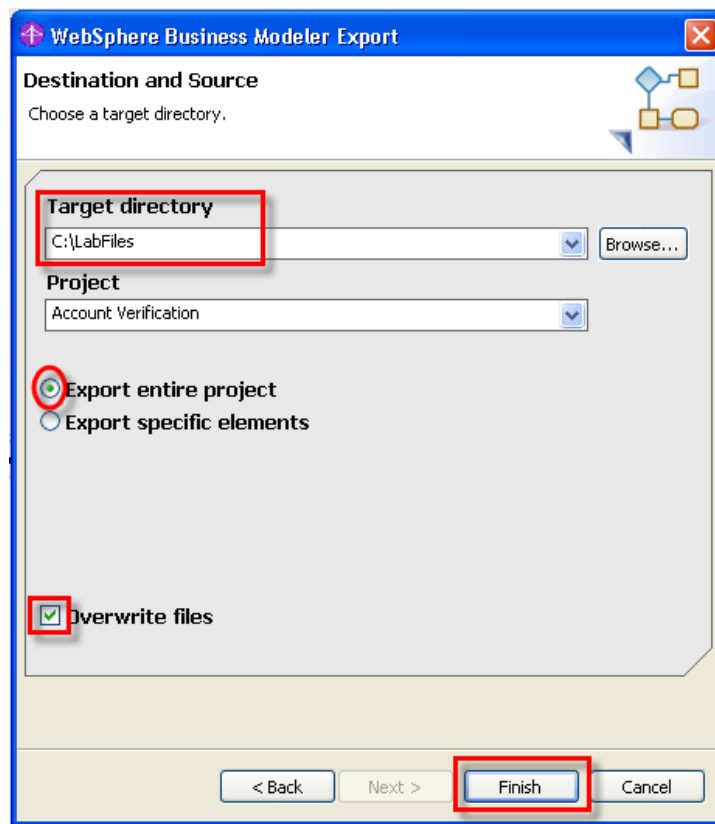
__5.   If you get an overwrite warning, click **Yes to All**.

__6.   The Export finished successfully message will appear after a few seconds. Click **OK**.
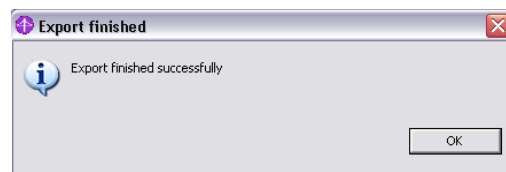
__7.   Two types of files have been created:

   __a. A preliminary Monitor Model (.mm) file. This is the preliminary version of the Monitor Model that the I/T specialist will work with in the next phase to complete the definitions needed for WebSphere Business Monitor. This file contains all the KPI, Metric, Dimension and Alert definitions you created.

   __b. The graphical process diagram files (.svg files) that will be available for WebSphere Business Monitor to display in a Diagram view on the dashboard; you have seen a version of this diagram used in the Diagram views in the dashbaords.
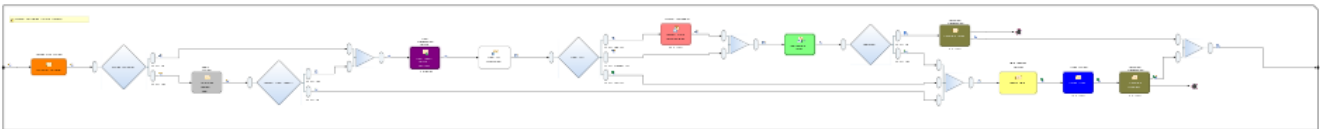
__8.   Click **File** > **Exit** to close WebSphere Business Modeler.
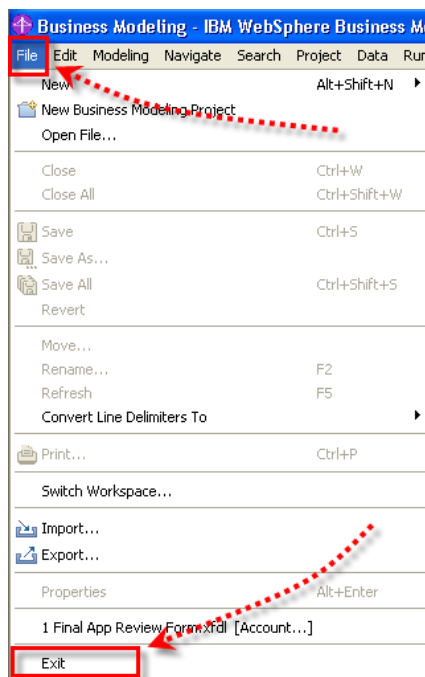
__9.   You have completed this lab.

Appendix A. IBM TechWorks

TechWorks is a worldwide team serving IBM software sales, technical, and development professionals and through these organizations our customers and partners. The TechWorks organization works to:

___iii.   Provide implementation, execution, and expertise for brand and general technical sales strategies and objectives

___iv.   Provide deep technical and product knowledge directly to customers and partners

___v.   Create material to teach our field teams, our partners, and our customers about what IBM products can do

___vi.   Operate world-class, high-quality facilities that support and enhance the interactions between our field technical sales team and our clients and partners

For more information about IBM TechWorks, talk with your IBM sales representative.

Appendix B. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental. All references to fictitious companies or individuals are used for illustration purposes only.

Appendix C. Trademarks and copyrights

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	WebSphere	Lotus

Adobe, Acrobat, Portable Document Format (PDF), and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Other company, product and service names may be trademarks or service marks of others.