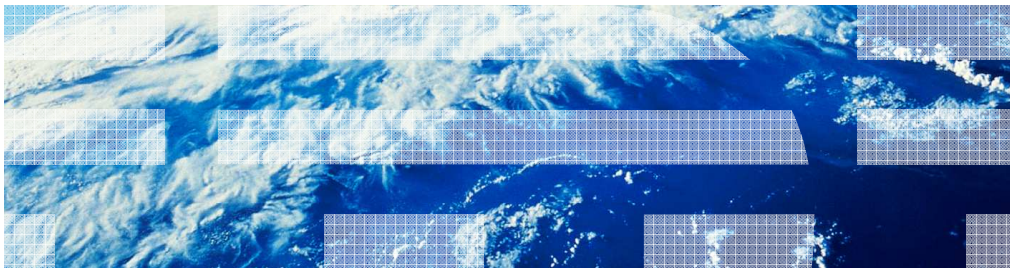




Business Process Management

IBM Business Process Manager V7.5

Application development process



© 2011 IBM Corporation

This presentation provides information about developing process applications using IBM Business Process Manager V7.5.

Table of contents

Goals

- Introduce two methodologies that can be used when developing business process applications
- Provide information about how and when to use key features of the Business Process Manager that are designed to help manage the development of business process applications

- Methodologies
- Basic approach
- Broader approach
- Roles
- Partitioning the work
- Version management
 - Creating and using tracks



The goal of this presentation is to introduce two basic methodologies that are relevant to the development of business process applications. And, show how they can be used along with the features available with the IBM Business Process Manger suite, to create business process applications.

You will learn how to develop your business process applications in an iterative fashion and how to use some of the key concepts such as application partitioning and version management.

Methodologies



- Process Designer first (top down)
 - Good for new projects
 - Iterative discovery
 - Data are defined as the scenario develops
 - Use of *Coaches* to gather requirements
 - Data closely coupled to a given business process
 - Client interaction
 - Develop as you go
- Integration Designer first (bottom up)
 - Larger organization with existing infrastructure
 - Data / Schema already defined by architecture group
 - Must use existing definitions which are hard to change
 - Typically requires heavy use of maps and mediations
- Common to both
 - Always have a runnable skeleton of the process flow
 - Make use of the default Coaches and the Inspector



Green fields



Integration

There are two key development tools available with IBM Business Process Manager V7.5

There is the Process Designer, which excels at creating new human centric business processes and the Integration Designer which provides the tools for creating automated services and integrating with existing applications and processes.

The Process Designer supports a top down development process whereas the Integration designer supports a bottom up approach.

When developing a business process application using the Process Designer you will use an iterative approach to discover and gather the requirements.

With the Process Designer, there are Coaches which are instrumental in gathering the requirements and defining the data.

When a new activity is placed in the flow, it will have a 'default Coach'. When run in the Inspector, the default Coach will display basic information about the activity and will allow the flow to run without any additional work. It acts as a place holder and a point of discussion for defining the requirements. This facilitates the iterative development. Using this approach the data are defined as the flow and the activities are defined. The data definitions are closely coupled to the business processes and the activities. With this methodology, there is a high degree of client interaction and the business process is developed as the requirements are gathered. There will always be a running business process, although it might not be fully functional.

With large enterprises, particularly one that is well on the way to adopting a Service Oriented Architecture, there is typically an existing infrastructure composed of reusable assets or services. In this case the interfaces and the data structures are already defined and managed by an architecture group. This means that the interface and data definitions are tightly controlled and difficult to change. Which in turn means that in order to integrate the reusable parts into a business process you will need to use data mapping tools and mediations. This is where the Integration Designer shines.

Both work together to provide the ultimate solution.

When using either approach it is a good practice to provide stubs for the services based on the interfaces so that the business process can run in the Inspector even though the implementation is not complete.

Basic approach

- In **interactive sessions** with the appropriate *stakeholders*
- Create a Process Application
- Create a Business Process Definition
 - Create the flow using default Human Services
 - Identify roles, assign them to swim lanes and move the activities into the appropriate swim lanes
 - Create participant groups
 - Identify existing services to use.
 - Identify data required for the activities and services
 - Develop and refine the Human Services and integration services
- Keep doing this until you have met the exit criteria
- Deploy to test
- Deploy to QA
- Deploy to production



Development of a business process begins with gathering the requirements from the stakeholders.

Using the Process Designer you can hold interactive sessions with the stakeholders and layout the basic flow using the default human service, and identifying who does what, and when.

As you determine who is doing the work, you identify the roles, create the swimlanes and then define the participant groups. Along the way you will also identify the need for some automated services. You can use the default human service as a place holder until the service or a stub becomes available.

Walking through the flow of activities and services with the stakeholders, the discussion shifts to the data required at each activity. Using the Process Designer, the data elements and structures are easily created and added to the business process flows.

Once the data requirements are known, you can start to develop the Coaches to be used with the Human Services.

As you continue the interactive sessions with the stakeholders you continually refine the business process definitions and the coaches.

With each iteration you can verify that the requirements are being met by running the business processes with the Inspector.

When you are satisfied that the business requirements are met, you have a runnable business process that is 80% to 90% functional.

Upon completion of the services, you are ready to move the process application to the test environment, then quality assurance and finally to production.

Broader approach

- Knowing what you do about the basic approach
 - Identify the processes that are to be developed
 - Make basic decisions about the User Interface
 - Testing using Business Space
 - Will custom Business Space widgets be needed
 - This might not be known until a couple of iterations into the process
 - Determine if there are going to be *Advanced Integration Services* used
 - Plan on packaging them into **toolkits**
 - Recruit integration developers
 - Identify the common components
 - Target them for **toolkits**
 - Identify key performance metrics
 - Create version management plan
 - Assign teams and individuals to be responsible for the Process Application, Toolkits and services
- Define the key interfaces between the components
- Stub out what you can in order to allow the business process to flow in the Inspector
- Iteratively develop the components using the basic approach
- When ready deploy to test
 - Develop scripts for deploying
 - Automate tests where possible
- When ready deploy to QA, then to Production

Once you know how it works, you can step back and think about other aspects of the development process

Think in terms of components and toolkits

Always think about what data will be useful for monitoring and where to gather it

Now that you have an idea of the basic approach, you are probably thinking about the broader picture.

What is the final application going to look like? What will the user-interface be like?

Will you be using the IBM Business Space? Will the Coaches and Business Space provide what you need or will you need to create custom Widgets?

What about reusable components? Will you be using Advanced Integration Services? How will these be packaged? How will you keep the developers working in their own components, toolkits or process applications?

All of these questions and more will surface very quickly.

Therefore, knowing what you know about the key concepts, capabilities and the basic development approach, you need to broaden the development process to include these.

Identify the processes that are to be developed. Which ones need to be done first.

Decide on the user interface framework you are going to use. The Business Space or home grown.

Determine if you are using Advanced Integration Services. If so then make sure you have developers with the skills. Plan on packaging them as Toolkits.

Identify any common components, and plan on packaging them as Toolkits.

Assign teams and individuals to be responsible for the toolkits and other components of the process application.

Identify key performance metrics and tracking points so that you can measure the performance of the process application.

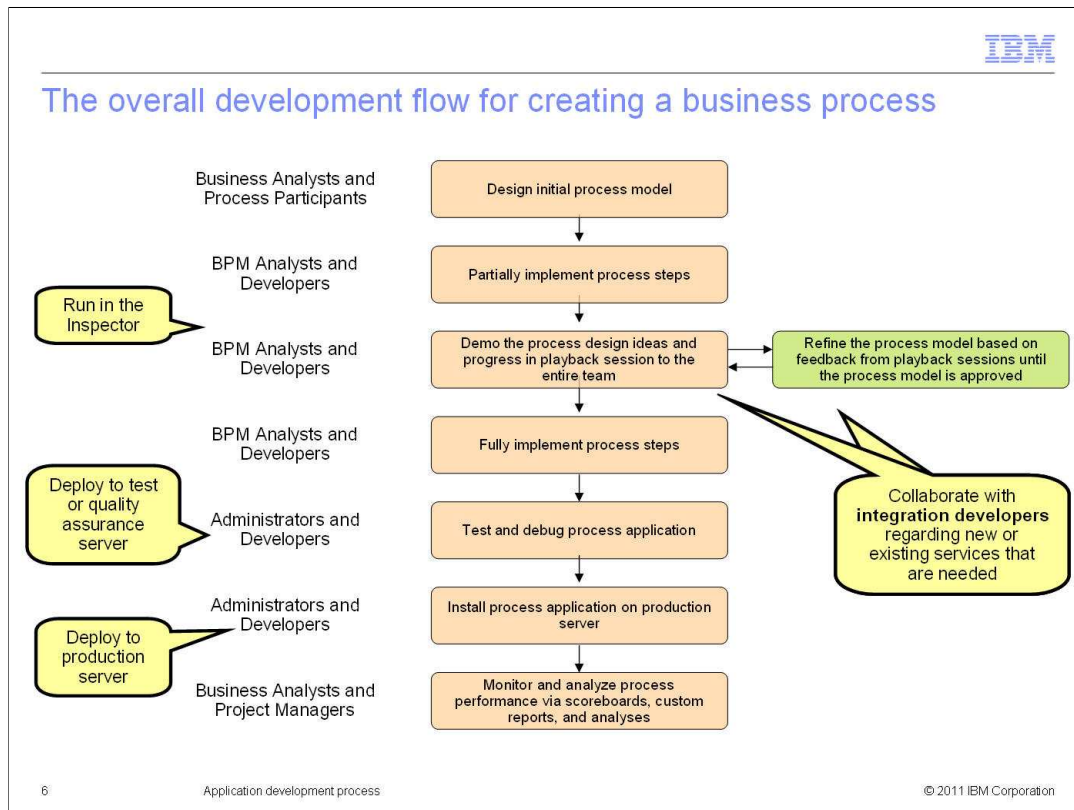
Create a version management plan that outlines how you will service the process application once it is deployed.

Define the key interfaces between the components.

Stub out what you can in order to allow the business process to flow in the Inspector.

Iteratively develop the components using the basic approach.

The overall development flow for creating a business process



Here a graphical representation of the development process.

This picture also shows the roles of the people that are involved at each step.

Notice the “refine the process model” step in green. This is where the iteration happens.

Development roles

- Process Developer
 - Person that develops business process flows using Process Designer
- Integration Developer
 - Person that develops integration services using Integration Designer
- Lead Developer
 - Very familiar with both roles above
 - Understands the version management
- Administrator
 - Manages users and privileges
 - Manage deployments
 - Test, QA and Production
 - Understands the Process Center and WebSphere Process Server administration
 - Knows how to configure LDAP for WebSphere Process Server



This is a simplified view of the roles needed during development.

A business process developer uses the **Process Designer** to create, edit and test business processes.

A service or integration developer uses the **Integration Designer** to create, edit and test services that integrate disparate parts of the enterprise.

The lead developer or team leader needs to be very familiar with both roles and they need to understand the Process Center version management system.

The Administrator must understand the administrative models for both the Process Center and the WebSphere Process Server. Many of the higher level functions are available through the Process Center and the Process Admin console. When it comes to details of installation, deployment, debug and configuration, there is also a need to know and understand the WebSphere Process Server.

For complex environments, it will also be necessary to know how to configure LDAP for the WebSphere Process Server.

Partitioning the work

- The Process Application or Toolkit is the level of management
- Toolkits are for components that are to be **shared** between different Process Applications or Business Processes
- Within a Process Application or Toolkit each business process application or set of services should have clearly defined owners
- Snapshots are read-only versions of a Process Application or Toolkit
 - When you start to edit one, you are editing the tip
- Only **snapshots** of *toolkits* can be shared
 - for example, referenced by other Toolkits or Process Apps
- Process Applications cannot be shared
- The responsibility / ownership model will keep the developers from overwriting each others work
- By factoring the shared components into Toolkits, the dependent developers can reference version A of the Toolkit while the Toolkit developers work on version B of the Toolkit

Partitioning the work is really the process of identifying the components, their interfaces and what developers are responsible for them.

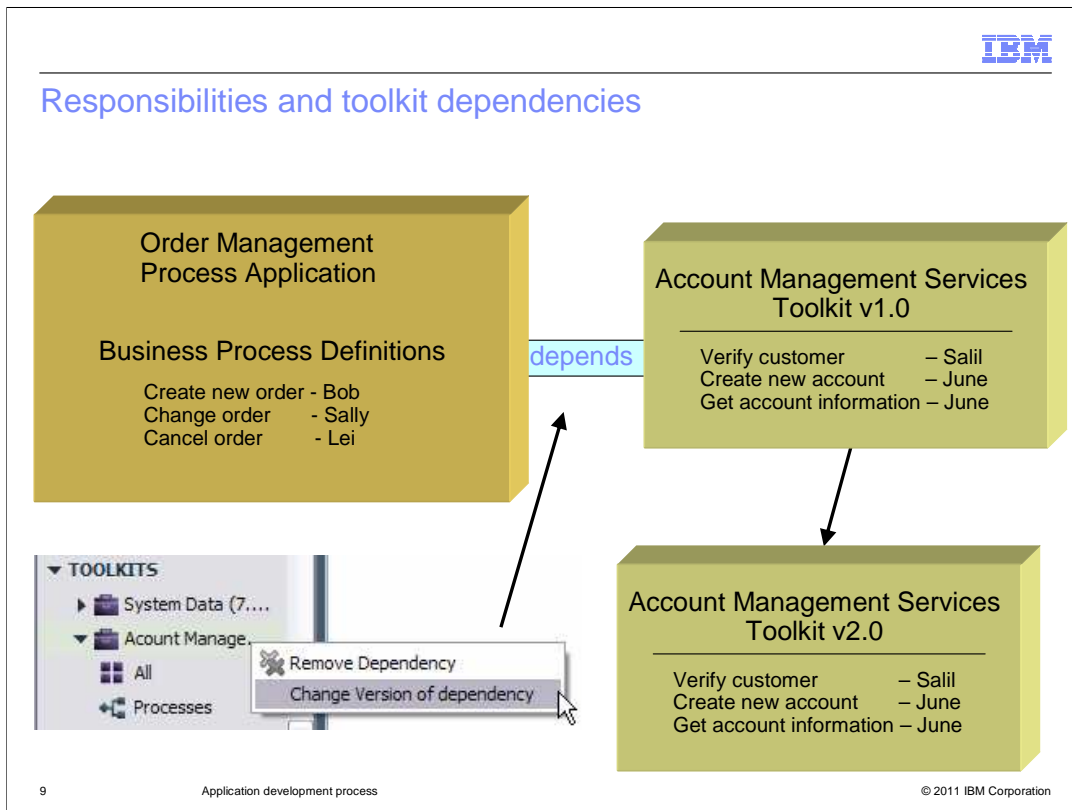
If the work is partitioned correctly the different teams can work on their components independently without impacting the other teams or developers.

When considering this task, keep in mind that the unit of management is the Process Application or Toolkit. If you have components that are to be shared they need to go into a toolkit. A toolkit snapshot can be created to allow other teams to use the existing function while the toolkit team continues to develop the newer functions. The authors of a toolkit can be managed so that only the toolkit team can make changes to it.

Always remember, Process Applications cannot be shared.

Within a Process Application the individual business process cannot be versioned so it is up to you to manage what developers work on a particular business process. If a process developer has write authority to a given Process Application, then they will also be able to change any of the business processes in that process application. This is why it is important to define who is responsible for a given part of the Process Application, and manage it closely.

Responsibilities and toolkit dependencies



Here you see the Order Management Process Application that depends on the Account Management Services Toolkit V1.0.

The toolkit developers, Salil and June, delivered version 1.0 which gave, Bob, Sally, and Lei the key functions they needed with the rest stubbed out.

It gave them what they needed to continue developing the Order Management Process Application.

Meanwhile the toolkit development team can continue working on the toolkit, developing version 2.0. When the new version is ready, then the process application team can easily change the dependency to the new version, or change is back if there is a problem.

Version management

- Version management comes in two flavors
 - Process applications, deployed from the repository in IBM Process Center
 - Enterprise applications deployed directly from IBM Integration Designer
- The focus in this presentation is on process applications
- A **snapshot** is an immutable version of a Process Application or Toolkit
- The **tip** is the current working 'editable' copy
 - It is automatically created when you try to edit a snapshot
 - When work is completed and ready to share or promote, a new snapshot is created
- Tracks can be enabled for a given Process Application or Toolkit
 - Enables you to create new branches from a given snapshot
 - Picture or screen capture on the next slide
- There are no 'diff' tools for comparing two snapshots
- Artifacts from Integration Designer that are under CVS or Rational ClearCase control are
 - Linked by way of meta data

Advanced
configuration
only

With the Advanced configuration of Business Process Manager, there are two development tools, the Process Designer and the Integration Designer. Each has a different version management mechanism. If you only work in the Integration Designer and do not add your Advanced Services to Process Applications, then you can continue to use the version management that is provided with the Integration Designer.

The moment you associate your Advanced Service components to a Process Application or Toolkit, they are managed by the Process Center. When this happens they become a part of the Process Application or Toolkit and from thereafter are versioned as part of the Process Application or Toolkit snapshot.

The Process Center version management mechanism uses snapshots. One or more developers opens a process application in the Process Designer and starts working on it. All current work is done on what is called 'the tip', like the tip of a branch. When you feel like you have a significant milestone that you want to save, you create a snapshot. The snapshot is an immutable version of 'the tip' at a specific point in time.

When you get to the point in your development that you need to have parallel development from the same snapshot, you can enable the "tracks" feature. Tracks are parallel branches from a common snapshot. Tracks are disabled by default and you can enable the feature or disable it at anytime.

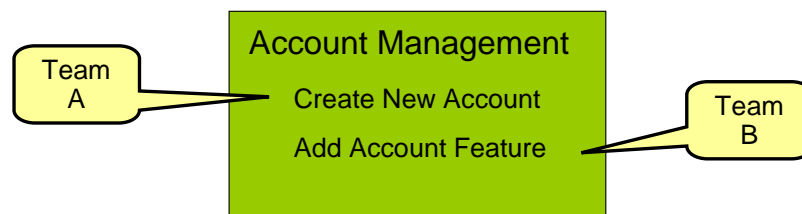
One very important thing to remember about using "Tracks", to merge the parallel tracks, you will have to manually copy the elements to a common branch. This is another reason why it is important to have the developer roles and responsibilities clearly defined. This is discussed more in the next couple of slides.

When working in the Integration Designer, you can also use a source code control system, such as CVS or Rational Clear Case,

for your Advanced Integration Service components. When you associate your, 'source code controlled', service with a Process Application or Toolkit, the meta-data for the source code control system is updated to indicate the association. Therefore, after making the association you should commit the changes to the source code control system.

Version management – using tracks

- Why Tracks?
 - Tracks allow
 - parallel development to occur with isolation from changes in other tracks
 - Example:
 - tracks enable one team to fix the current version of a process
 - » while another team builds a completely new version based on new external systems and a new corporate identity
 - Example:
 - you have two teams working on different business process definitions (BPD) within the same Process Application



11

Application development process

© 2011 IBM Corporation

Tracks allow for parallel development. Here are two examples where tracks can be used.

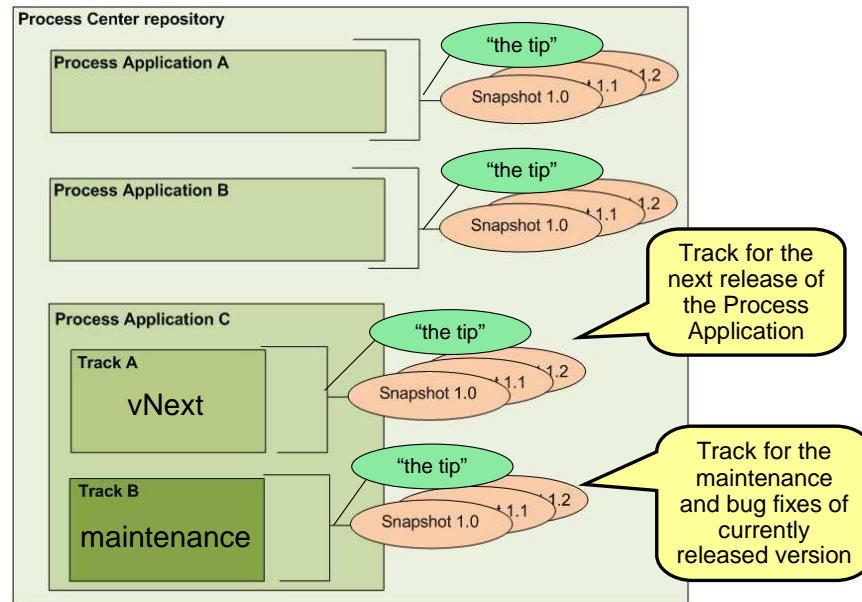
The first example is the classic new development versus maintenance tracks. They both start from a common snapshot and then diverge.

Any fixes that need to be incorporated into the new development track, must be manually copied in.

The second example is where there is a single Process Application, with many business process definitions that can be developed in parallel.

In this case one or more developers work on the 'Create New Account' business process definition while different business process developers work on the "Add Account Feature."

Parallel development – example 1



12

Application development process

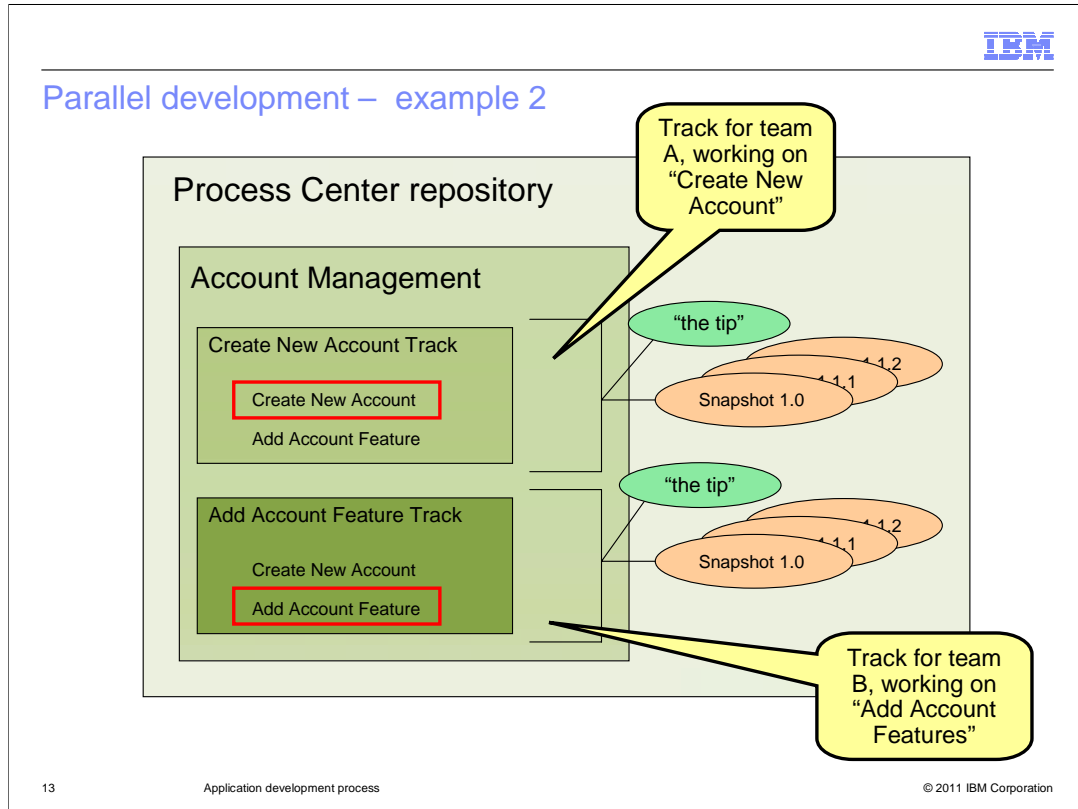
© 2011 IBM Corporation

In this graphical example the first thing to notice is that everything is contained in the Process Center repository.

Next, you'll notice that tracks are contained within the Process Application. You will experience this containment when you start working with tracks.

A track is created from a snapshot and once it is created a new snapshot, with the new name is created. There is an example of creating and working with snapshots in subsequent slides.

Parallel development – example 2

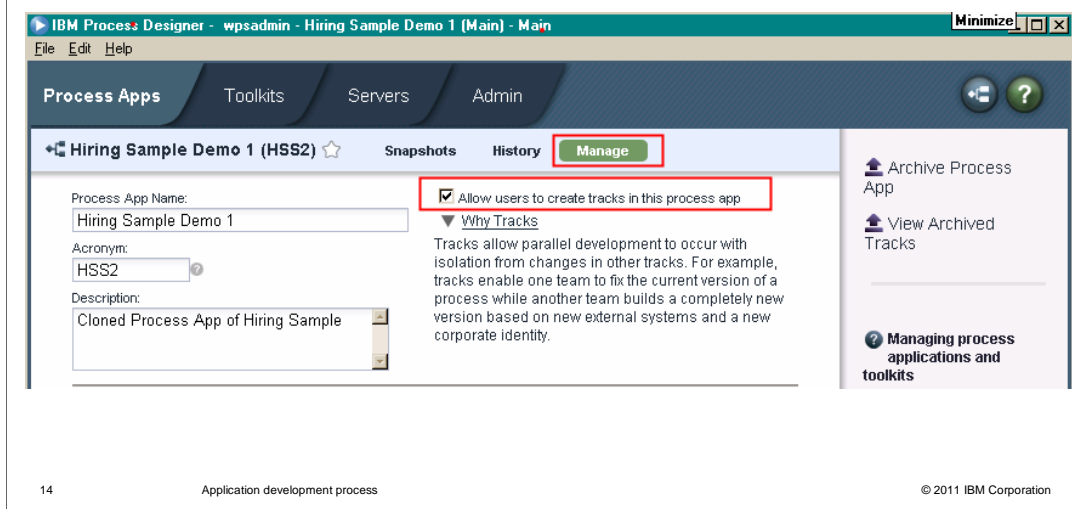


This is a graphical representation of the second example, where there are parallel development tracks within the Process Application.

When it is time to merge the two tracks, you can use one of the existing tracks as the 'master' and copy the other into it, or you can copy both into the original track called 'main.'

Enabling tracks

- Tracks must be enabled for each Process Application or Toolkit before they can be used



Here you see the view within the Process Center where you enable or disable the track feature. Select the Process Apps tab, then the process application, then the Manage tab.

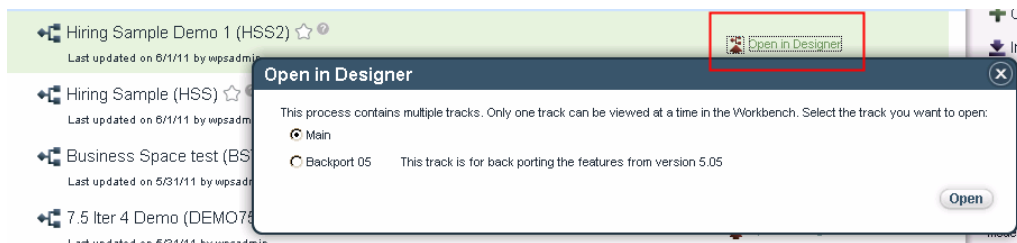
Creating a track

The screenshot displays the IBM Process Designer interface. At the top, the title bar reads "IBM Process Designer - wpsadmin - Hiring Sample Demo 1 (Main) - Main". Below this is a menu bar with "File", "Edit", and "Help". The main navigation area includes "Process Apps", "Toolkits", "Servers", and "Admin". The current view is for "Hiring Sample Demo 1 (HSS2)", with tabs for "Snapshots", "History", and "Manage". A "Sort Snapshots By:" dropdown is set to "Date", with filters for "All", "Deployed", and "Archived". A snapshot named "SSS1" is listed, created on 6/1/11 by wpsadmin, and is not yet deployed. A red box highlights the "New Track" option in the context menu. The "Create New Track" dialog box is open, showing the following fields: "New Track Name:" with the value "Backport 05", and "Description:" with the text "This track is for back porting the features from version 5.05". A "Create" button is at the bottom right of the dialog. The footer contains the page number "15", the text "Application development process", and the copyright notice "© 2011 IBM Corporation".

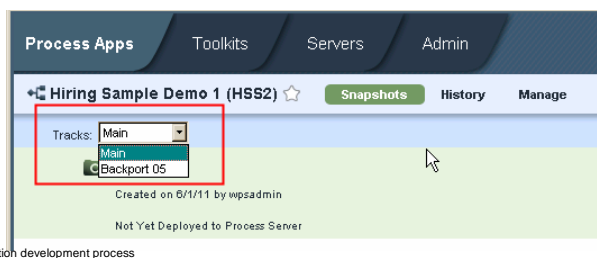
Once the track feature has been enabled, a new menu item is displayed with the operations available for snapshots. Selecting the 'New Track' option will display the 'Create New Track' dialog. Enter a name for the track and a description. The unique acronym is created for you automatically and you won't see it.

Using a track

From the top level you'll get a prompt



From inside you'll have a drop-down list



16

Application development process

© 2011 IBM Corporation

When you open a process application that has tracks created, you will see the “Open in Designer” dialog shown in the top screen capture. Here you’ll be able to select which track you want to open in the Process Designer.

If you are already in the Process Application and you want to switch to another track, there is a drop down menu that will let you make your selection. This is shown in the screen capture at the bottom.

Version management – naming context

- A *version context* is a combination of acronyms that uniquely describes a process application or toolkit
 - a business process definition (BPD) in a process application is typically identified by the process application acronym, the snapshot acronym, and the name of the business process definition
- Each type of acronym has a naming convention
- The acronym is limited to a maximum length of seven characters from the [A-Z0-9_] character set
 - except for the snapshot acronym, which can also include a period.
- The acronym is seen in the WebSphere Process Server Admin Console, Process Center, Process Admin, and logs

 Hiring Sample Demo 1 (HSS2) ☆ ?

Last updated on 6/1/11 by wpsadmin

<input type="checkbox"/>	CNA02-Tip	
<input type="checkbox"/>	DAT01-Tip	
<input type="checkbox"/>	DNA02-Tip	

Business Level Applications in the WebSphere Process Server console

17

Application development process

© 2011 IBM Corporation

The next few slides dive into some technical detail that is helpful when you start to test and debug your application. It is good to know how it works before you get too deep into your project.

Every Process Application and Toolkit must have a unique acronym that is used to identify it in the Process Center repository and in the server runtime. When versions are introduced, there is a relationship between the Process Applications that stem from the same root. To identify a particular version there is the version context. The version context is a combination of the process application acronym, the snapshot acronym, and the name of the business process definition.

The rules for creating acronyms are show on the slide. It is important to understand the acronym because it will show up when you are debugging you process applications. The acronym is seen in the WebSphere Process Server Admin Console, Process Center, Process Admin, and the logs.

The screen capture shown here is a snippet of the “Business Level Applications” section of the WebSphere Process Server Admin console.

Snapshot acronym

- Snapshot acronym
 - If the snapshot name meets the criteria for a valid snapshot acronym, the snapshot name and acronym is the same
 - When using the **mediation flow component** version-aware routing function,
 - name your snapshot so that it conforms to the `<version>.<release>.<modification>` scheme (for example, 1.0.0)
 - because the snapshot acronym is limited to seven characters,
 - the digit values are limited to a maximum of five total digits (five digits plus two periods)
 - therefore, care should be taken when the digit fields are incremented, because anything beyond the first seven characters is truncated

This is note of warning if you are using the version-aware-routing function of the mediation flow component.

You must adhere to the version dot release, dot modification, scheme when naming the snapshot.

Watch out for the name length limitation. Anything beyond the first seven characters are truncated.

Track acronym

- Automatically generated from the first character of each word of the track name
 - Example: a new track created with the name **My New Track** will result in an acronym value of **MNT**
- The default track name and acronym are **Main**
- Deployment to a IBM Process Center server includes the track acronym in the version context if the track acronym is not Main

As mentioned previously. The acronym for a track is created for you automatically. It is derived from the name of the track, tacking the first character of each word. When you create your first track, the original track is named 'main'. This is the default track, which can be changed at anytime.

When deployed to a Process Center Server, the track acronym is included in the version context if the track is not the main track.

Summary

- Methodologies
- Basic approach
- Broader approach
- Roles
- Partitioning the work
- Version management
 - Creating and using tracks



In this presentation you were introduced to two basic process development methodologies that are relevant to developing business processes with Business Process Manager. The top-down and bottom-up. The primary distinction between the two being where the interfaces are created and managed.

With the top-down approach, the interfaces belong to the process developers and are created in the Process Designer.

With the bottom-up approach, the interface belongs to the integration developers and are created in the Integration Designer.

Both approaches are fully supported by the Business Process Manager Advanced configuration.

With the Standard configuration, a bottom-up approach is still possible if the services are web services. In this case the interfaces belong to the web service provider.

You also learned how some of the key features and concepts, such as version management, partitioning the work, and development roles, can influence how you approach your development process.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_BPMV75_DevProcess.ppt

This module is also available in PDF format at: ../BPMV75_DevProcess.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, ClearCase, Rational, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2011. All rights reserved.