



IBM Software Group

WebSphere® Process Server V6.1.2 WebSphere Integration Developer V6.1.2

Business process choreographer overview



@business on demand.

© 2008 IBM Corporation
Updated September 12, 2008

This presentation will provide an overview of the changes, enhancements, and new Business Process Choreographer features in WebSphere Process Server and WebSphere Integration Developer version 6.1.2.

Goals

- Introduction to BPC changes since V6.1
- Introduction to BPC changes in V6.1.2



The primary goal is to explain the changes, enhancements, and new Business Process Choreographer features introduced in V6.1.2. However, there were also some significant changes to the Business Process Choreographer in the first fix pack of V6.1. These will also be covered, in order to provide a complete picture of what has changed since the release of V6.1.

Agenda

- V6.1.0.1
 - ▶ Allow force completion of BPEL wait activities
 - ▶ More fine-grained common base events
 - ▶ Improved business process navigation performance
- V6.1.2
 - ▶ Improved monitoring support
 - ▶ Exploitation of business calendars
 - ▶ Graphical process widget
 - ▶ Business space for human workflow
 - ▶ Enhanced dynamicity for in-flight processes



This slide lists all of the changes, enhancements and new Business Process Choreographer features in WebSphere Process Server and WebSphere Integration Developer V6.1.2, in addition to those introduced in the first fix pack of V6.1. Each of these is described in detail in this presentation.

Section

New features in V6.1.0.1



This section will explain the changes, enhancements, and new Business Process Choreographer features introduced in the first fix pack of V6.1.

Allow force completion of BPEL wait activities

- In V6.1.0.0, an attempt to force complete BPEL wait activities fails with an error message

The screenshot displays the Business Process Choreographer Explorer interface. The main window shows the 'Activity' details for a 'Wait' activity. The 'Force Complete' button is circled in red, and a red arrow points from it to the error message below. The error message reads: 'CWVBU0012E: The state 'Waiting' does not allow this action. Verify that the state is one of the following: (Ready, Claimed, Stopped, Running) before triggering the action.'

Business Process Choreographer Explorer

Welcome admin | Logout | My Substitutes | Define Substitutes | Define Views | Customize | Help | About

Process Templates
My Process Templates

Process Instances
Started By Me
Administered By Me
Critical Processes
Terminated Processes
Failed Compensations

Task Templates
My Task Templates

Task Instances
My To-dos
All Tasks
Initiated By Me
Administered By Me
My Escalations

Activity

Use this page to view details about an activity and its associated process. ⓘ

Restart Force Complete

Activity Description

Activity Name Wait
Kind Wait
State Waiting
Description

Details Tasks Custom Properties

Activated 6/2/08 5:52:25 PM
Completed
Expires on 6/2/08 5:54:25 PM
Owner
Process Instance Name Test #4
Process Template Name TestProcess

Error

This page shows details of an error which occurred. [Search for more information.](#)

Exception Error Message Exception Stack Trace All Exception Error Messages Complete Exception Stack Trace

CWVBU0012E: The state 'Waiting' does not allow this action. Verify that the state is one of the following: (Ready, Claimed, Stopped, Running) before triggering the action.

Business process choreographer overview © 2008 IBM Corporation 5

The first fix pack of V6.1 introduced a change with respect to forcing the completion of a BPEL wait activity. In V6.1, an attempt to force complete a BPEL wait activity fails with an error message stating that the action is not allowed for an activity in the “Waiting” state. This restriction has been removed in first fix pack of V6.1, and now BPEL wait activities can successfully be forced to completion.

More fine-grained common base events

- Supporting event sequencing
 - ▶ Progress counter identifies the current navigation step in the set of all process steps of the process
 - It was previously always set to zero, but now in V6.1.0.1 the progress counter is set so as to identify the current navigation step
 - ▶ Local counter used to order events inside a transaction
 - ▶ These two counters can be used to put events into the correct order without the need to use the event timestamp
 - Avoids any problems related with timestamps in a distributed application
 - To sort events, look at the progress counters first and compare them...
 - If the progress counters of two events are the same, then compare the local counters



The next pair of enhancements in the first fix pack of V6.1 provide for more fine-grained common base events. The first set of new events was added in support of event sequencing. The progress counter identifies the current navigation step in the set of all process steps of the process. It was previously always set to zero, but now the progress counter is set so as to identify the current navigation step. The local counter is used to order the events inside a transaction. Together these two counters can be used to put events into the correct order without the need to use the event timestamp. Doing so avoids any problems related with timestamps that can occur in a distributed application (for example, time zone differences). Two events can be sorted by first looking at the progress counters and comparing them. If the progress counters of the two events are the same, then compare the local counters.

More fine-grained common base events

- Supporting custom properties
 - ▶ New events are generated when a custom property is set
 - ▶ Events only generated for **non-binary** custom properties
 - ▶ For the activity-level custom property set event, the event correlation sphere identifiers indicate the context in which the custom property has been set
 - If the custom property is set using an API request...
 - the event correlation sphere identifiers are set as for a process event
 - If the custom property is set in a Java™ snippet...
 - the ECSCurrentID is set to the activity instance ID of the Java snippet and the ECSParentID is set to the process instance ID



The second set of new events was added in support of custom properties. New events are generated when a custom property is set (one each for a process, activity, human task, and escalation instance). The new events are generated only for non-binary custom properties. For the event that is generated when an activity-level custom property is set, the event correlation sphere identifiers indicate the context in which the custom property has been set. This makes it possible to determine if the custom property is set by an API request or in a Java snippet.

Improved process navigation performance

- Traditional JMS based navigation for business processes involves chained transactions with persistent JMS messages

- ▶ After each transaction, the thread context is thrown away, and cannot be used for optimizations

- A new, lightweight navigation style, called WorkManager based navigation, has been introduced in WebSphere Process Server 6.1.0.1

- ▶ WorkManager threads are used for thread creation
- ▶ Server affinity allows re-use of cached objects
- ▶ The quality of service is the same as in JMS based navigation
- ▶ Communication to the database is reduced
- ▶ Performance improvement in extreme cases ~ factor 2

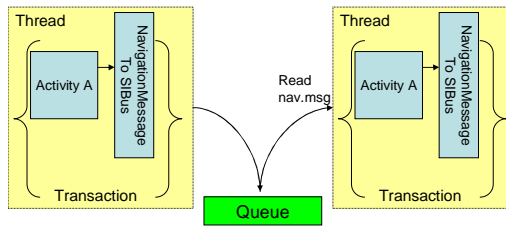
Note that for 6.1.0.1, i-Fix IZ21335 must be applied in a cluster installation if WorkManager based navigation is used.



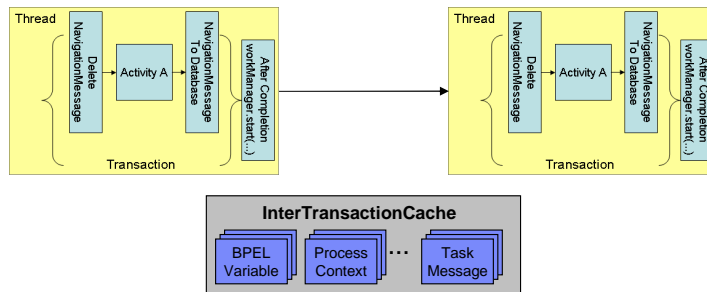
The last significant change that was introduced in the first fix pack of V6.1 provides for improved process navigation performance. Before this change, business process navigation involved chained transactions with persistent JMS messages. After each transaction, the thread context is thrown away and cannot be used for optimizations. The first fix pack of V6.1 introduces a new, lightweight navigation style, called WorkManager based navigation. It is so called because it is based on the WorkManager function provided by WebSphere Application Server. The WorkManager function is documented in the WebSphere Application Server 6.0 Information Center, but basically it is a mechanism for dispatching and monitoring asynchronous work using asynchronous beans. In this new process navigation style, WorkManager threads are used for thread creation. Server affinity allows re-use of cached objects. The quality of service is the same as in JMS based navigation, but communication to the business process choreographer database is reduced, resulting in performance improvements of roughly a factor of two in extreme cases.

Improved process navigation performance

JMS based navigation



WorkManager based navigation



This slide compares the traditional JMS based navigation (shown at the top of the diagram), with the new WorkManager based navigation (shown at the bottom of the diagram). Notice the use of a JMS queue for storing thread context using persistent JMS messages in the traditional JMS based navigation, compared to the inter-transaction cache which is used in the new WorkManager based navigation.

Improved process navigation performance

- Enabled by a BFM custom property
 - Other custom properties used to configure behavior

The screenshot shows the IBM Integrated Solutions Console interface. The main content area displays the 'Application servers > server1 > Business Flow Manager > Custom Properties' configuration page. A table lists several custom properties:

Select	Name	Value	Description
<input type="checkbox"/>	RALTemplateName	BPC_CLIENT	The RAL template to be used by Process Choreographer
<input type="checkbox"/>	allowPerformanceOptimizations	no	Enables performance optimizations
<input type="checkbox"/>	interTransactionCache_size		Number of database objects that are kept in cache between

A yellow callout box points to the 'allowPerformanceOptimizations' property, stating: "Default is JMS based navigation. Set to 'yes' for WorkManager based navigation." The property name 'allowPerformanceOptimizations' is circled in red in the original image.

The new WorkManager based navigation is enabled by a Business Flow Manager custom property called “allowPerformanceOptimizations”. It can be set in the administrative console, along with other custom properties that can be used to configure the behavior of the new WorkManager based navigation. See the WebSphere Process Server V6.1 Information Center for more information on the custom properties that affect the performance and behavior of the Business Flow Manager.

Section

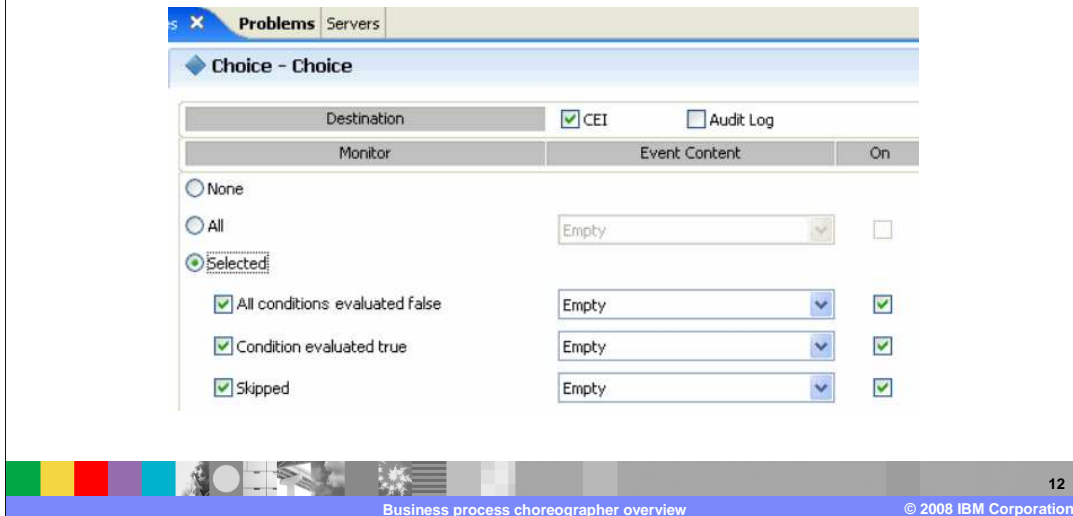
New features in V6.1.2



This section will explain the changes, enhancements, and new Business Process Choreographer features introduced in V6.1.2.

Improved monitoring support

- New common base event generated for performed branch in a switch



The first new Business Process Choreographer feature introduced in V6.1.2 is improved monitoring support. This was implemented by allowing new common base events to be generated for the performed branch in a switch (which is also referred to as a choice activity). These new events allow for a business process to be monitored in greater detail than before. The picture shows the properties panel of a choice activity in the WebSphere Integration Developer, where you can specify that you want the new events to be generated.

Exploitation of business calendars

- Based on the sophisticated calendars in Modeler
- Created in WebSphere Integration Developer and deployed to WebSphere Process Server as an artifact similar to a data type or interface
- Can be used in BPC anywhere where simple and WebSphere calendars are currently allowed
 - ▶ BPEL wait activity timeout
 - ▶ BPEL invoke activity expiration
 - ▶ Human task activity expiration
 - ▶ Human task over-due duration
 - ▶ Human task escalation notification



The first significant new Business Process Choreographer feature introduced in V6.1.2 is called exploitation of business calendars. Business calendars are created in WebSphere Integration Developer and deployed to WebSphere Process Server as an artifact similar to a data type or interface. They are based on the sophisticated calendars in WebSphere Modeler, and they allow you to define periods of time that should be included or excluded when calculating durations. BPC exploitation of business calendars allows business calendars to be used anywhere where simple and WebSphere calendars are currently allowed. This includes BPEL wait activity timeout, BPEL invoke activity expiration, human task activity expiration, human task over-due duration, and human task escalation notification.

IBM Software Group IBM

Exploitation of business calendars

1. Create new business calendar

2. Define necessary time intervals

3. Specify business calendar on timeout

Business process choreographer overview © 2008 IBM Corporation

This slide shows how to create and use a new business calendar in WebSphere Integration Developer. The first step is to create a new business calendar. This can be done in the **Business Integration** pane by right-clicking on **Business Logic** and selecting the **New -> Business Calendar** menu item. The business calendar editor is then displayed. Use the business calendar editor to define periods of time to be included or excluded from consideration. To add a time interval to be included, click on the **Add Interval** button. Likewise, to add a time interval to be excluded, click on the **Add Exception** button. Once all the necessary time intervals have been defined in the business calendar, the new business calendar can be used when defining a duration. In this example, the new business calendar is used to define the duration of a BPEL wait activity.

One important thing to keep in mind when defining the time intervals of a business calendar is that by default all time is to be excluded, except for those time intervals which are explicitly included. Thus, a calendar which only excludes certain time intervals is considered to be not valid. If you want to define a business calendar by specifying only time intervals to be excluded, you must also add one interval to be included. This interval would be defined so as to encompass all of time (that is all day, every day). Note that time intervals to be excluded are sometimes referred to as “OffTime”, whereas time intervals to be included are sometimes referred to as “OnTime”.

Graphical process widget

- General purpose graphical representation of business processes and human tasks
 - ▶ Used in business space for human workflow and BPC explorer
 - ▶ Multiple detail levels depending on usage scenario
 - ▶ Pop-up menu on activities for triggering of actions
- Replacement for SVG based process views
 - ▶ No browser plug-in required
 - ▶ Using JavaScript and Dojo Toolkit
 - ▶ Improved viewer capabilities
 - More fine-grained zooming
 - Better layout algorithm



The next significant new Business Process Choreographer feature introduced in V6.1.2 is the graphical process widget. This widget provides a general purpose graphical representation of business processes and human tasks. It is part of two different client offerings. It is used in the new Business Space for Human Workflow (which is covered later in this presentation), and in the BPC Explorer, where it is a replacement for the current SVG based process viewer. The new graphical process widget has many benefits over the current SVG based process viewer with respect to process viewing. First and foremost is the fact that, unlike the current SVG based process viewer, the new graphical process widget does not require a browser plug-in. Instead, it uses JavaScript and the Dojo Toolkit. Also, it has improved viewing capabilities such as more fine-grained zooming, and a better layout algorithm. But what truly sets the new graphical process widget apart is its interactive nature. Depending on the usage scenario, the widget can display the same process at multiple levels of detail, from the low-level BPEL details, to a high-level view showing only the human tasks involved in the business process. Finally, the new graphical process widget can be used to trigger actions on activities in the process using pop-up menus.

Graphical process widget

Overview

The screenshot displays the Business Process Choreographer Explorer interface. On the left, there is a sidebar with navigation options: Process Templates, Process Instances, Task Templates, and Task Instances. The main area shows the 'Process State View' for a process instance named 'Wgprprocess'. Above the graphical flowchart, there are two sliders: 'Detail level' and 'Zoom'. A callout 'Zooming' points to the Zoom slider. Another callout 'Various detail levels' points to the Detail level slider. A third callout 'Trigger actions using pop-up menu' points to a pop-up menu on the 'Initialize Complex Types' activity in the flowchart. The bottom of the screenshot shows a footer with 'Business process choreographer overview' and '© 2008 IBM Corporation'.

This picture shows the new graphical process widget within the BPC Explorer. Above the graphical view of the process you see two sliders. The slider on the right is used to zoom the graphical view in or out. This controls the amount of space occupied by the graph. The slider on the left is used to adjust the level of detail being displayed. This controls whether nodes in the graph are hidden or displayed. Finally, on top of the graph you see a pop-up menu which is used to trigger actions on activities in the business process.

Graphical process widget

- Multiple levels of detail

The image displays three screenshots of the Business Process Choreographer graphical process widget, illustrating different levels of process detail:

- Complete BPEL mode:** The leftmost screenshot shows a highly detailed BPEL process flow with many nodes and connections. A yellow callout bubble points to it with the text "Complete BPEL mode".
- Reduced BPEL mode:** The middle screenshot shows the same process flow but with many details hidden, resulting in a simpler, less complex graph. A yellow callout bubble points to it with the text "Reduced BPEL mode".
- Human Task only mode:** The rightmost screenshot shows only the human tasks (represented by person icons) involved in the process, with all automated BPEL logic hidden. A yellow callout bubble points to it with the text "Human Task only mode".

At the bottom of the slide, there is a blue bar with the text "Business process choreographer overview" on the left and "© 2008 IBM Corporation" on the right. The number "17" is also visible in the bottom right corner of the slide area.

This slide shows the three levels of process detail that the graphical process widget can display. On the left is the complete BPEL mode, where every detail of the BPEL process is displayed. In the middle is reduced BPEL mode, where some of the details of the BPEL process have been hidden to reduce the complexity of the graph. Finally, on the right is the human task only mode. This is the highest level of detail, showing only the human tasks involved in the BPEL process. All of the automated portions of the BPEL process have been hidden, reducing the graph to that of a human workflow.

Business space for human workflow

- New, ready-to-run, Web 2.0 based BPM client for business users
 - ▶ Rich UI widgets supporting human-centric client spaces
 - Predefined space layout available
 - Customized layouts can be defined
 - ▶ Graphical process widget for human workflow view
 - ▶ Predefined task templates for collaboration
 - Ask question
 - Request review
 - Request approval



As was mentioned previously, in addition to the BPC Explorer, the new graphical process widget is also used in a brand new client called the Business Space for Human Workflow. This is a new, ready-to-run Web 2.0 based BPM client for business users. It uses rich UI widgets that support human-centric client spaces. There is a predefined space layout available, but customized layouts can also be defined. Furthermore, it is intended that spaces and their layouts can be customized by business users, not just the IT staff. In addition to the graphical process widget for viewing human workflow, there are predefined task templates for common collaboration tasks. This includes the ability to ask a question of a co-worker, request that a work item be reviewed, or request that a work item be approved.

Business space for human workflow

■ Default space layout

The screenshot displays the Business Space for Human Workflow interface. Key components and callouts include:

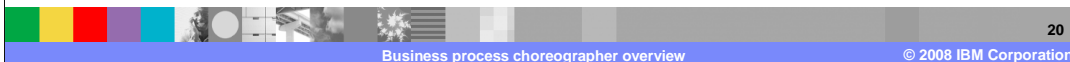
- Integrated Forms:** A callout pointing to the 'Task Information' and 'Task Creation' panes, which contain various input fields and forms for task management.
- Start a Task:** A callout pointing to the 'Task Creation' pane, indicating the area where new tasks are initiated.
- Human Workflow Process View:** A callout pointing to the central 'Human Workflow Diagram', which shows a graphical representation of the workflow process.
- Manage Individual and Team Tasks:** A callout pointing to the 'My Tasks' and 'Available Tasks' panes, which allow users to manage and select tasks.
- See list of team:** A callout pointing to the 'Team List' pane, which displays a list of available teams.

Business process choreographer overview © 2008 IBM Corporation

This picture shows the default space layout for the Business Space for Human Workflow. In the upper right frame there is a widget for starting a task. In the upper left is a widget which allows the business user to select an individual or team task to work on. In the middle of the browser window there are two panes which change depending on the selected task. The top frame has a widget which allows the business user to work on the selected task. In this particular case it is displaying a Lotus Form that has been generated for the task. Below that is the graphical process widget that you saw before, set to the highest level of detail possible, showing only the human tasks involved in the process.

Business space for human workflow

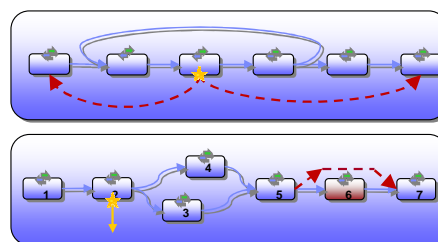
- Exploits business space manager
 - ▶ Common base framework for user interfaces in BPM products
- Based on WebSphere Process Server REST APIs
 - ▶ Provided by BPC and business rules
- Implementation uses Dojo Toolkit and JavaScript
 - ▶ To create iWidget compliant widgets
 - ▶ Runs in the user's browser
 - ▶ Code is served to browsers by business space manager



Underneath the covers, the Business Space for Human Workflow exploits the Business Space Manager, which is a common base framework for user interfaces in various WebSphere Business Process Management products. As such, many of the products in the WebSphere Business Process Management portfolio also offer Business Spaces which are relevant to their particular product domain. For instance, WebSphere Business Monitor provides a Business Space for monitoring business processes. Business Space for Human Workflow is based on WebSphere Process Server REST APIs that are provided by the BPC and business rules components. The implementation uses the Dojo Toolkit and JavaScript to create iWidget compliant widgets. These widgets run in the user's browser, and they are served to browsers by the Business Space Manager.

Enhanced dynamicity for in-flight processes

- Provides greater flexibility for predefined processes
- Allows adapting a process to a changed situation
- Enables solution administrators and business users to override the navigation of a process
 - ▶ Jump forward and backward between activities in a running process
 - ▶ Skip specific activities within a running process
 - ▶ Incorporate process relevant data changes in-flight
- Graphical process widget in BPC Explorer provides the user interface for dynamic modifications

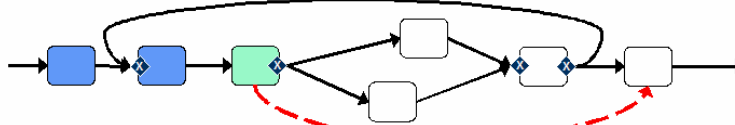


The last significant new Business Process Choreographer feature introduced in V6.1.2 is called enhanced dynamicity for in-flight processes. It provides greater flexibility for predefined processes, and allows for easily adapting a business process to changing business situations. This new support in WebSphere Process Server enables business users and solution administrators to override the navigation of a process by jumping forward and backward between activities, skipping specific activities, and modifying process relevant data in-flight. Out-of-the-box support for these dynamic modifications is provided by the BPC Explorer using the new graphical process widget. Alternatively, business users can directly benefit from these capabilities through custom business user clients. For example, “skip” or “return request” buttons, which use the same APIs as used by the new graphical process widget, can be added to custom business user clients. Business transactions that are dynamically adjusted in any of the previously mentioned ways are still traceable and measurable for retrospective investigations (for example, monitoring, and processing history). To allow for audit, respective monitoring events are generated upon dynamic modification (skip, jump, update variable).

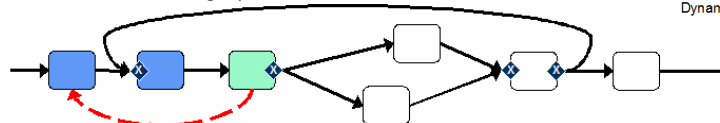
Enhanced dynamicity for in-flight processes




Jump between activities of a process

Jump forward within an in-flight process



Jump backward within an in-flight process



Current activity: 
 Predefined control flow: 
 Dynamic modification: 

Supported with new APIs

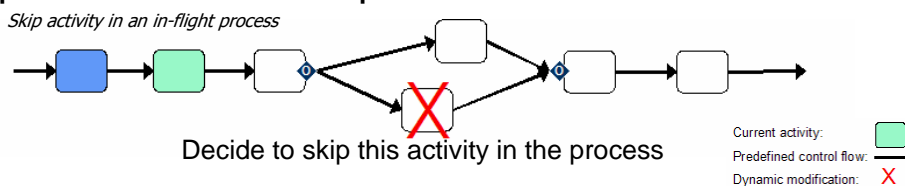
- *completeAndJump()* – completes activity and jump to target activity
- *forceCompleteAndJump()* - Force completion of activity and jump
- *skipAndJump()* - Skips current activity and jumps to specified target activity



To support dynamic modifications to in-flight processes, WebSphere Process Server V6.1.2 introduces jumping forward and backward between activities within a running process. Jumping from one activity to another is possible within a sequence and a cyclic flow (sometimes referred to as a single-threaded graph). However, it is not possible to jump between parallel activities, that is, within constructs where parallel activities are possible. The source activity of a jump must be a basic activity (for example, a human task, or an invoke activity). It must also be in one of the “running” states, that is, *ready*, *claimed*, *running*, *stopped*, or *waiting*. The target activity of a jump can be either a basic activity or a structured activity. Jumps are supported on the same level, that is, no jumps into nested constructs or to activities in the surrounding activity. Jumping backward is allowed, but it does not involve compensation. Three new Business Flow Manager EJB APIs enable jumps between activities within a running process. The *completeAndJump()* operation applies to human task activities and is used to complete the current human task activity and jump to the specified target activity. The complete semantics are the same as those supported in the current WebSphere Process Server release, except that the human task must not be completed with a fault. The *skipAndJump()* operation applies to human task activities, invoke, receive, wait, or any basic activity in the *stopped* state. It is used to skip the current activity and jump to the specified target activity. The *forceCompleteAndJump()* operation applies to any “running” basic activity, and is used to force the completion of the current activity and jump to the specified target activity. The new common base event *JUMPED* is emitted to track a jump from a source activity to a target activity. The event carries information about the target activity template name, and the user ID of the requestor. This event is emitted in addition to any other related events. For example, in the case of *completeAndJump()*, events related to both the complete and the jump are generated.

Enhanced dynamicity for in-flight processes

- Skip activities within a process

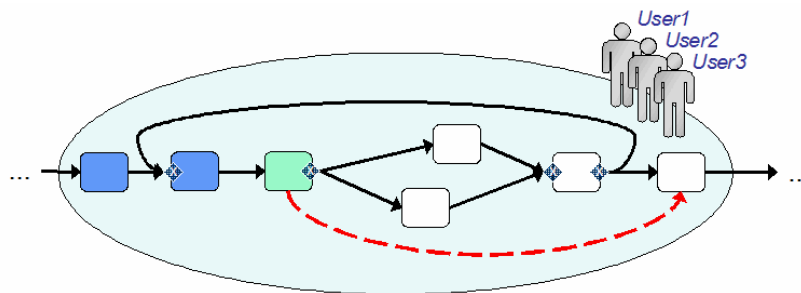


- ▶ New API *skip()*
- Skip can be combined with jump
 - ▶ New API *skipAndJump()*
- Undo skipping of an activity
 - ▶ New API *cancelSkipRequest()*

To support dynamic modifications to in-flight processes, WebSphere Process Server V6.1.2 introduces skipping an activity that is not needed for a particular in-flight process. This helps in dynamic human workflow scenarios and to support case handling scenarios. Support for skipping activities applies to either the current activity, or an activity that has not yet been reached in the process. The current activity can be skipped immediately as long as it is in a non-terminal state (for example, *running*, or *ready*). Future activities can be marked "*skipRequested*" so they are skipped when/if they are reached. An activity to be skipped needs to be a basic activity (for example, human tasks, invoke activity...). Structured activities (for example scope, sequence...) can not be skipped. The activity to be skipped can be identified by activity instance ID, or by activity name (as long as the name is unique). Thus, for example, skipping an activity in parallel activities/flows is supported; however, in a parallel *forEach* construct, identification by activity name is ambiguous, therefore skipping is not possible. Within a flow, upon explicitly skipping an activity, the follow-on navigation continues with evaluating the transition conditions of links leaving the skipped activity. (Note: this is not to be confused with "implicit" skipping of activities that happens upon dead path elimination.) A skip request is applicable to a single navigation of the process activity instance. That is, after an activity has been skipped, another skip request needs to be issued if the same activity is to be skipped should it be reached again. As with jumping, there are new Business Flow Manager EJB APIs that enable skipping an activity within a running process. The *skip()* operation is used to skip the specified activity immediately if it is in a "running" state (that is, *ready*, *claimed*, *running*, *waiting*, or *stopped*). The *skip()* operation can also be used to skip a future activity, in which case the activity is marked to be skipped, that is, its *skipRequested* property is set to *true*. Note: it is not checked whether the activity to be skipped can be reached again. If the activity to be skipped is in a "running" state, all that has been done so far on behalf of this activity is ignored/reset/terminated. For a human task activity, this means that the corresponding task instance is terminated. For an activity implemented by a child process, this means that the child process is terminated (and compensated if compensation was specified). If there is a timer scheduled (for example, for expiration of an invoke, or for a wait activity), it is cancelled. The *cancelSkipRequest()* operation is used to cancel a previously requested skip request. If an activity instance has been marked for skip, that is, its *skipRequested* property has been set to *true*, the property is reset to *false*. The new common base event *SKIP_REQUESTED* is emitted if an activity is marked for skip, or if a request for skipping is cancelled. The event carries information about the activity template name, and the user ID of the requestor. The new common base event *SKIPPED_ON_REQUEST* is emitted if an activity is being skipped upon explicit request (as opposed to a skip that happens in case of dead path elimination). Skipped activities must not lead to process failure (for example, due to missing process data/variables). Therefore, failures in expressions that access process data/variables result in stopping the process instance rather than leading to an overall process failure.

Enhanced dynamicity for in-flight processes

- Authorization for dynamic modifications
 - ▶ Authorization is possible for selected sets of activities (for example that constitute *business cases*) within a process
 - New authorization concept on scope level
 - Specific business users and administrators can be authorized to initiate jumps, or skip certain activities within the scope they are authorized for



The enhanced dynamicity features introduced in WebSphere Process Server V6.1.2 required a change to the authorization model for process administration. In version 6.1 and earlier releases, administrative tasks can be assigned to a process, or to specific activities. Now, an administrative task can also be specified at the scope level. This allows assigning "stakeholders" to a selected set of activities, the set being defined by the activities contained in the scope. Such stakeholders can apply dynamic modifications to in-flight processes, such as skipping activities, or jumping between activities. Process administrators are also allowed to perform dynamic modifications of in-flight processes. Introducing scope-level authorization, however, allows for finer-grained authorization specification in addition to what has been supported so far. Authorization for a particular set of activities does not require being process administrator. Note that being an activity administrator is not sufficient authorization for being allowed to jump between activities.

Enhanced dynamicity for in-flight processes

- Update or modify variables of a process
 - ▶ set process variables while process instance is running
 - ▶ useful in repair scenarios where inconsistent data needs to be changed
- Supported with new APIs
 - ▶ *getAllVariableNames()* - returns the names of all variables
 - ▶ *getInitialVariableValue()* - retrieves the initial value of the specified variable
- BPC Explorer supports updating variables
 - ▶ via the *activity instance details* page
 - ▶ variables that are visible from a specific activity can be updated

25

Business process choreographer overview

© 2008 IBM Corporation

Besides jumping between activities within a running process and skipping an activity that is not needed for an in-flight process, the third aspect of enhanced dynamicity is the ability to update or modify the variables of a process. This can be business information associated/carried through the process, like customer name, address, credit rating, and so on. But it can also be data that is created during the process, controlling process logic/execution (like a flag that controls a "retry" of an activity or the like). This is useful in repair scenarios where inconsistent data needs to be changed. This capability is supported by new Business Flow Manager EJB APIs. The *getAllVariableNames()* operation returns the names of all variables, and the *getInitialVariableValue()* operation retrieves the initial value of the specified variable. The variable names and their initial values can be retrieved in any execution state of the activity instance. The caller must have at least reader authority for the associated process instance. The user interface for dynamic modification of process variables is provided by the BPC Explorer. This is done on the *activity instance details* page where any variables that are visible from a specific activity can be updated.

Section

...and in conclusion...



The next slide provides a summary of what you have learned during this presentation.

Summary

- V6.1.0.1
 - ▶ Allow force completion of BPEL wait activities
 - ▶ More fine-grained common base events
 - ▶ Improved business process navigation performance
- V6.1.2
 - ▶ Improved monitoring support
 - ▶ Exploitation of business calendars
 - ▶ Graphical process widget
 - ▶ Business space for human workflow
 - ▶ Enhanced dynamicity for in-flight processes



In this presentation you have been introduced to the major new Business Process Choreographer features in WebSphere Process Server and WebSphere Integration Developer V6.1.2. Some significant changes, enhancements, and new features introduced in the first fix pack of V6.1 were also introduced for the sake of completeness.

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 e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WBPMV612_BPC_NewFeatures.ppt

This module is also available in PDF format at: [../WBPMV612_BPC_NewFeatures.pdf](..WBPMV612_BPC_NewFeatures.pdf)



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

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM WebSphere

A current list of other IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

Java, JavaScript, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2008. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

