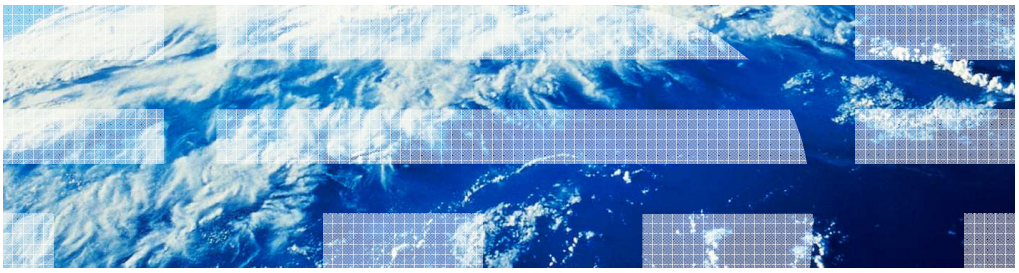
Business Process Management
IBM Business Process Manager V8.0.1

Coach data validation



IBM Business Process Manager version 8.0.1 introduces a server side validation framework for validating coach data that is entered by a user at runtime. This presentation will give you an overview of the new coach data validation framework.

Overview

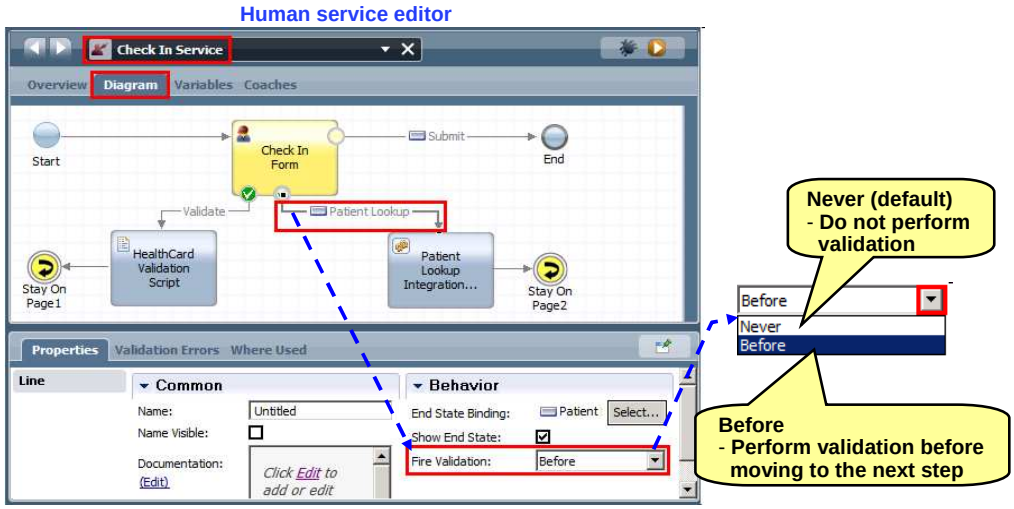
- Easier modeling of coach data validation using a service or a server script
- Can use any service as a validation service except Ajax service and human service
- Only one validation service or server script per coach to validate it's data
- Multiple coaches can use the same service or server script in a human service diagram
- New system variable coachValidation of type CoachValidation to contain error data

With the new coach data validation framework, you can easily model validation logic that identifies invalid user input and notifies the user at runtime. Validation logic is defined using a server script or a service. Any service, other than an Ajax service or a human service, can be used as a validation service. A coach can only use one validation service or server script to validate its data. Multiple coaches in a human service flow can use the same validation service or server script.

Business Process Manager version 8.0.1 provides a new system variable called coachValidation, which is of type CoachValidation, to contain information about validation errors. The CoachValidation business object is defined in the Coaches 8.0.1 toolkit.

When can coach data be validated?

- Before the flow moves from a coach to the next step in a human service flow
- New property “Fire Validation” on the link that connects a boundary event to the next step in a human service flow



In a human service flow, a boundary event in a coach is used to move the processing from the coach to the next step. Typically a Button stock control is used to trigger the boundary event. But you can define any coach view to trigger a boundary event.

In version 8.0.1, you can define the validation to be invoked on a particular boundary event. A new property called “Fire Validation” is added to a link that connects a boundary event to the next step to indicate if the validation should be invoked for this boundary event.

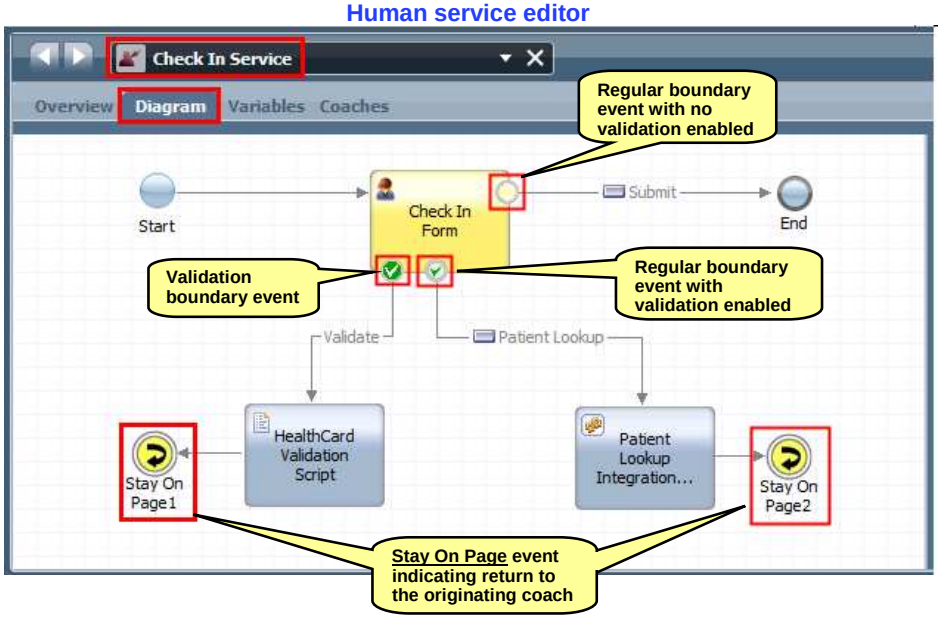
The property has two predefined values “Never” and “Before” to choose one from. The value “Never” is a default value which means validation should not be performed for the coach data. The value “Before” signifies that the coach data should be validated before the flow proceeds to the next step in a human service flow.

The runtime validation described here is different than any client side validation that might be defined for a given coach control. For example, if you define minimum and maximum values for a Decimal control and a user inputs a value outside the range at runtime, a validation error is reported. This client side validation occurs as soon as the data is input in a field and is not associated with any boundary event.

The next slide will describe the new elements that you see here in the screen capture.

New diagram elements

Human service editor



Version 8.0.1 visually represents boundary events on coach nodes in a human service flow. The screen capture on this slide shows examples of the types of boundary events on the Check In Form coach node.

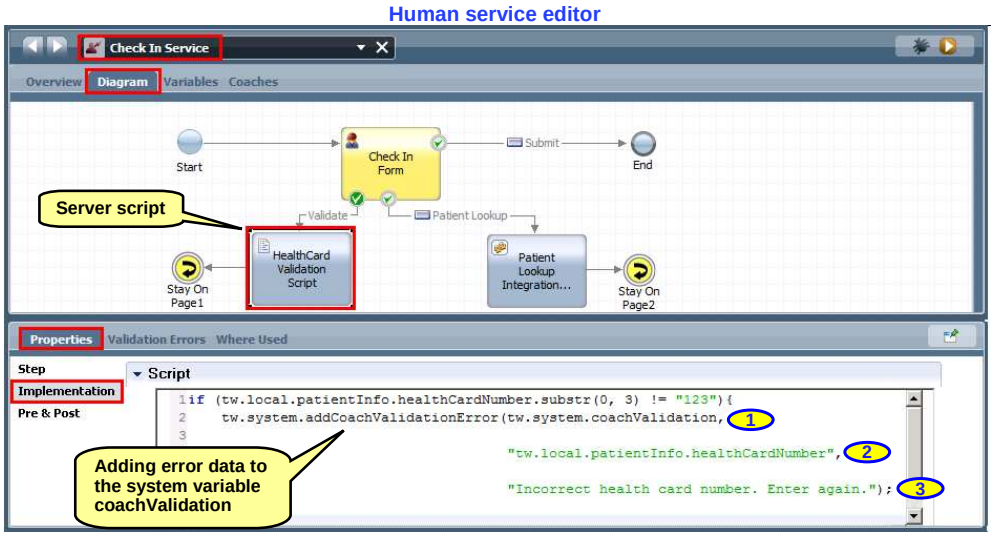
A boundary event without validation enabled is represented by a hollow circle, as shown at the top right corner of the node.

When you set the Fire Validation property value on a boundary event to Before, two things happen. First, a green check mark is shown inside the boundary event icon, as shown in the center of the bottom of the node. Second, a validation boundary event is added to the coach node. A validation boundary event is represented by a solid green circle with a white check mark inside, as shown at the bottom left corner of the node. You need to connect this validation boundary event to the server script or validation service that implements the validation logic.

Version 8.0.1 also introduces a new Stay On Page event in a human service flow. Connecting to a Stay On Page event causes the flow to return to the originating coach. You will typically link the validation server script or validation service to this event so that, when there is a validation error in the input data, the user is returned to the originating coach to correct the input data.

The Stay On Page event is not limited to validation purposes. Anytime you want to return the user to the originating coach after performing some action, instead of creating a loopback connection to the coach, you can use a Stay On Page event. In the example on this slide, the Patient Lookup Integration service looks up patient data and then displays the patient data on the originating coach Check In Form. Hence, the Patient Lookup Integration service is connected to the Stay On Page2 event.

Validation using a server script



1. Parameter 1 – System variable coachValidation to contain error data
2. Parameter 2 – Full path to the data item with an error in a String format
3. Parameter 3 – Error message to be displayed to a user in a String format

This slide illustrates how to define validation using a server script.

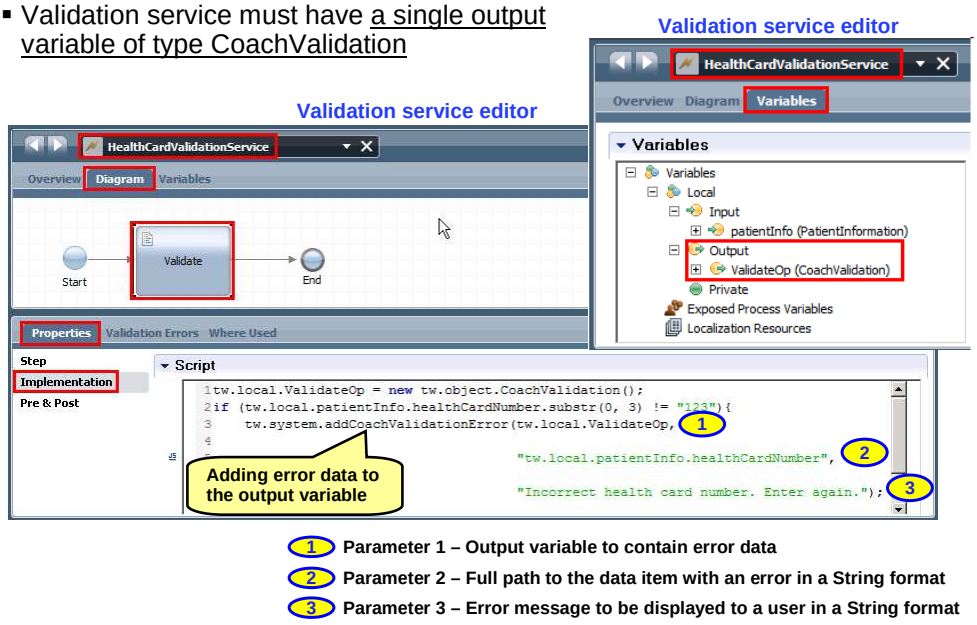
You have to specify validation logic on the Implementation page of the Properties view for the server script. When you write a validation rule, add the error data to the system variable coachValidation using an API such as tw.system.addCoachValidationError. This API takes three parameters. The first parameter is the system variable coachValidation. The second parameter is a string containing full path to the data item that contains the error. And the third parameter is the message that should be displayed to a user if this error occurs.

The example code shown here checks whether the health card number entered by a user is valid and, if not, adds the error data to the system variable coachValidation.

The tool provides a few more APIs that you can use to clear all validation errors from a given coachValidation instance, to remove errors for a given data item path and update error message for a given data item path.

Validation using a service (1 of 2)

- Validation service must have a single output variable of type CoachValidation



1. Parameter 1 – Output variable to contain error data
2. Parameter 2 – Full path to the data item with an error in a String format
3. Parameter 3 – Error message to be displayed to a user in a String format

Here is an example of how to define validation using a service.

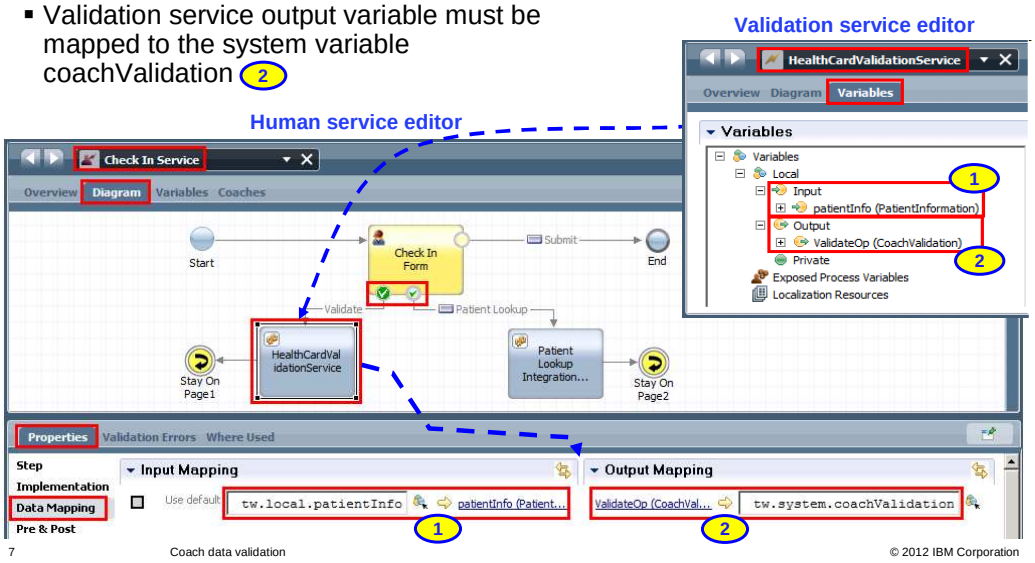
A validation service must have only one output variable and it must be of type CoachValidation. When you specify the validation logic, add the error data to the output variable using an API such as tw.system.addCoachValidationError. This API takes three parameters. The first parameter is the output variable of the service. Note that this is different than the previous slide, which uses the system variable coachValidation. The second parameter is a string containing full path to the data item that contains the error. And the third parameter is the message that should be displayed to the user if this error occurs.

The example on the slide shows a service named HealthCardValidationService. The Variables page of the service editor is shown in the screen capture at the top right. The service has a single output variable named ValidateOp, which is of type CoachValidation. The Diagram page of the service editor and the Properties view are shown in the screen capture on the left. The example code is the same as the previous slide except for the first parameter of the tw.system.addCoachValidationError API call.

The next slide describes how to use this service in a human service flow.

Validation using a service (2 of 2)

- Human service variable that holds the data being validated must be mapped to the validation service input variable that has the same name and type (1)
- Validation service output variable must be mapped to the system variable coachValidation (2)



When you add a validation service to a human service flow diagram, you have to specify the input and output data mapping, just like you do for any other service.

As part of the input mapping, the human service variable containing the data that's being validated must be mapped to the validation service's input variable with the same name and type. Look at the places highlighted on the slide with the number 1. The Variables page of the HealthCardValidationService editor is shown in the screen capture at the top right. The service has an input variable called patientInfo of type PatientInformation. The screen capture at the bottom shows the Diagram page of the Check In Service editor and the Data Mapping page of the Properties view for HealthCardValidationService element. In the Input Mapping section, the variable patientInfo of Check In Service is mapped to the variable patientInfo of HealthCardValidationService. The validation will not work properly if you do not follow this input mapping requirement.

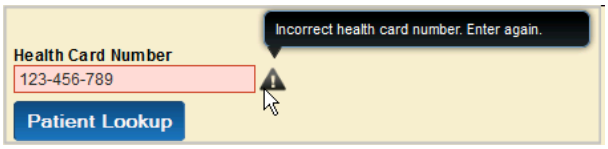
As part of the output mapping, the output variable of the validation service must be mapped to the system variable coachValidation. Look at the places highlighted on the slide with the number 2. The HealthCardValidationService has an output variable called ValidateOp, as shown in the screen capture at the top right. It is mapped to the system variable coachValidation in the Output Mapping section, as shown in the screen capture at the bottom.

You are prevented from connecting a validation boundary event to a validation service if the data mapping is not specified for the validation service.

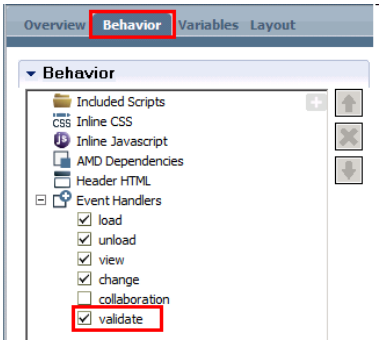
Runtime presentation

- Validation error displayed at runtime when the data with error is bound to a coach view
- Predefined runtime presentation of a validation error
  - Pink background color and black indicator
  - Hovering over the warning marker displays the error message defined by a coach developer
- Runtime error presentation can be customized for a coach view using the new validate event handler on the Behavior page of a coach view editor

Browser window



Coach view editor



8 Coach data validation

At runtime, when the user enters data that is invalid based on the validation logic, a validation error is displayed in the corresponding coach view.

But, if the invalid data is not bound to any coach view, then the validation error is not displayed.

The default visual representation of a validation error in a coach view is shown on this slide. The coach view is highlighted with a pink background and a red border. And a black, triangular indicator is shown beside the coach view. If the user hovers over the black indicator, the error message defined in the validation logic is displayed, as shown in the screen capture.

You can customize the visual representation of a validation error using the new validate event handler in the coach view editor. See the validate event handler implementation in the Text stock coach view for reference.

Summary

- Using a server script or a service, you can model coach data validation in a simplified manner
- Validation errors represented visually at runtime when data with error is associated with a coach view
  - Tool provided visualization used by default
  - Visualization can be customized by a coach developer by using validate event handler of a coach view

This presentation introduced you to the new coach data validation framework in version 8.0.1 of Business Process Manager. It gave you a simple way to model coach data validation using a server script or a service. You can notify users of data validation errors at runtime by visually displaying the error in the coach view.

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_BPM801_Coach_Data_Validation.ppt

This module is also available in PDF format at: ../BPM801_Coach_Data_Validation.pdf

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

Trademarks, disclaimer, and copyright information

IBM, the IBM logo, and ibm.com 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" at http://www.ibm.com/legal/copytrade.shtml

Other company, product, or service names may be trademarks or service marks of others.

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.