



IST-2103 STP ARTEMIS

"A Semantic Web Service-based P2P Infrastructure for the Interoperability of Medical Information systems"

SPECIFIC TARGETED RESEARCH PROJECT
PRIORITY 2.3.1.11 eHealth

An Addition to Artemis – D4.3.1 - Software tools for automated Web service discovery and composition exploiting service semantics:

ebBP Editor, User Guide (Draft v1.0.1)

DRAFT

Project Dates: Project Start Date: January 1, 2004
Project End Date: June 30, 2006
Project Duration: 30 months

Leading Contractor Organization: METU - SRDC

Development Team: Yavuz GURCAN, Veli BICER, Tuncay NAMLI, Mehmet OLDUZ, Bugrahan AKCAY, Ali DEMIR (XMLStylist – XSD Editor), Ozgur GULDEREN (CPP Editor), METU-SRDC

Document Authors: Yavuz GURCAN, Ali DEMIR, METU - SRDC

Short Description: This document describes the ebBP Editor of the Artemis platform.

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Table of Contents

1	Introduction	3
2	Role of the ebBP Editor	3
3	User Guide.....	4
3.1	Starting the ebBP Editor.....	4
3.2	ebBP Editor components	6
3.3	XMLStylist - XSD Editor	7
3.3.1	Abstract	7
3.3.2	Usage Areas.....	8
3.3.3	General View.....	8
3.3.4	General Usage	8
3.3.5	Validation	10
3.3.6	Limitations	11
3.4	Creating a Business Collaboration	11
3.5	Removing a Business Collaboration	12
3.6	Creating a Role.....	12
3.7	Creating a Business Transaction Activity	13
3.8	Creating a Collaboration Activity	16
3.9	Creating a Complex Business Transaction Activity	17
3.10	Creating a Success.....	17
3.11	Creating a Failure	17
3.12	Creating a Fork.....	17
3.13	Creating a Join.....	17
3.14	Creating a Decision	18
3.15	Specializing a Business Transaction	19
3.16	Creating a Business Transaction	21
3.17	Creating a Specialized Business Transaction.....	22
3.18	Editing the Process Specification	22
3.19	Saving the Process Specification.....	22
3.20	Importing another Process Specification (XInclude).....	22
4	Conclusion.....	22

1 Introduction

This document provides brief information about the ebBP Editor, and a User Guide for end-users presenting the usage and capabilities of the ebBP Editor. Please note that, this document DOES NOT provide a background on ebBP. Moreover, the ebBP Editor is based on the ebBP v2.0.1 specification, but is not dependent on this specification. That is, it can be easily adapted to future specification versions.

The ebBP Editor is a tool designed to help the user in creating generic as well as domain specific Business Process Specifications based on ebBP Version 2.0 by using semantic mechanisms. The editor lets the user to create Process Specifications from scratch or use the existing ones.

The ebBP Editor implementation is based on an XML Schema (XSD) Editor, which is used in creating the XML segments of ebBP process specification. This generic XSD Editor parses the XSD file and dynamically creates the graphical user interface of the corresponding element defined in the XSD. The user can then fill in the necessary fields by providing required information. The editor also provides the user with information about the necessity, cardinality and optionality of the fields. Once the user has filled in the information through the GUI, a validation check is made, and the information is serialized into XML.

The ebBP Editor also provides a graphical user interface to define domain specific business transaction classification hierarchies based on ebBP Business Transaction and the associated patterns. A user can specialize any of the generic ebBP Business Transaction patterns by specifying properties, or the document to be exchanged, but not necessarily a complete transaction. For this purpose, a basic ebBP ontology is dynamically extended. When the user finishes the specialization of a transaction template, the new transaction (template) is automatically annotated by a name given by the user.

When the user specializes a transaction annotated with a node of this ontology, a subclass is created with a name provided by the user, and the specialized transaction is annotated with it. Note that, the user has the ability to annotate this (template) transaction with the nodes of other domain-specific ontologies as well, since ebXML allows registry objects to be annotated with more than one classification hierarchy. The nodes of the ontologies introduced are then created in the ebXML registry as ClassificationNodes. Using these ClassificationNodes it is possible to search for specialized Transactions to be used as building blocks of Process Specifications.

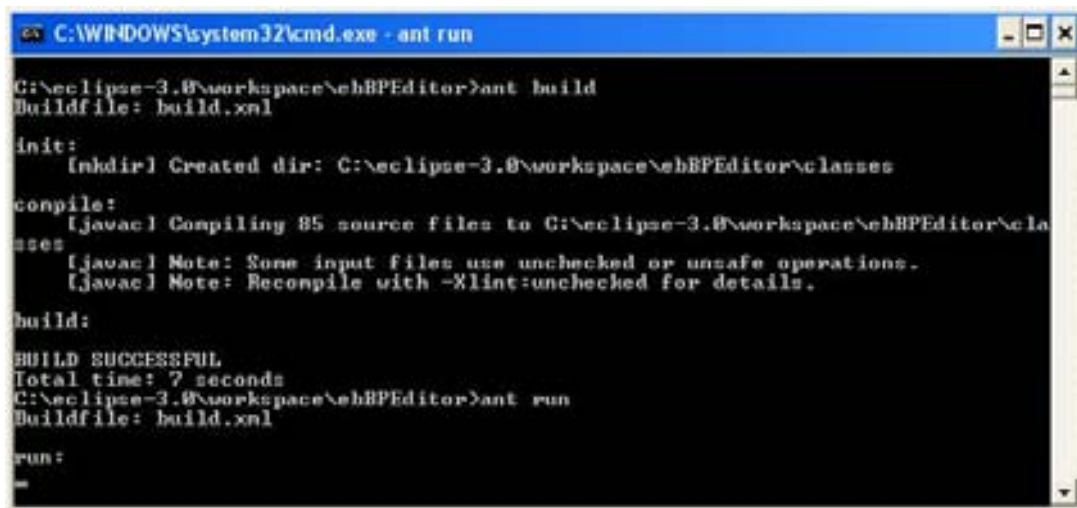
2 Role of the ebBP Editor

The ebBP Editor does not only facilitate the user in creating a Process Specification valid to the latest ebBP specification, but also allows domain-specific semantic annotation of Business Transactions. This semantic information will provide building blocks for creating a Business Process Specification Schema.

3 User Guide

3.1 Starting the ebBP Editor

The ebBP Editor can be compiled with the “ant build” command. After successfully compiling, the “ant run” command will start the editor (see Figure 1).



```
C:\WINDOWS\system32\cmd.exe - ant run
C:\eclipse-3.0\workspace\ebBPeditor>ant build
Buildfile: build.xml

init:
    [mkdir] Created dir: C:\eclipse-3.0\workspace\ebBPeditor\classes

compile:
    [javac] Compiling 85 source files to C:\eclipse-3.0\workspace\ebBPeditor\classes
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

build:
BUILD SUCCESSFUL
Total time: 7 seconds
C:\eclipse-3.0\workspace\ebBPeditor>ant run
Buildfile: build.xml

run:
```

Figure 1 - ant commands to start the ebBP Editor

Initially, the ebBP Editor will start together with the XMLStylist (XSD Editor, more information is available in section 3.3 about the XSD Editor) (see Figure 2). This is due to the fact that initially the editor starts with a Business Collaboration, and this Business Collaboration should be created with the XSD Editor.

The user is supposed to provide the name and name ID of the Business Collaboration at this stage. Of course it is also possible to provide other attributes as well. After creating the Business Collaboration and closing the XSD Editor, the Business Collaboration will be represented with a tab in the ebBP Editor (see Figure 3).

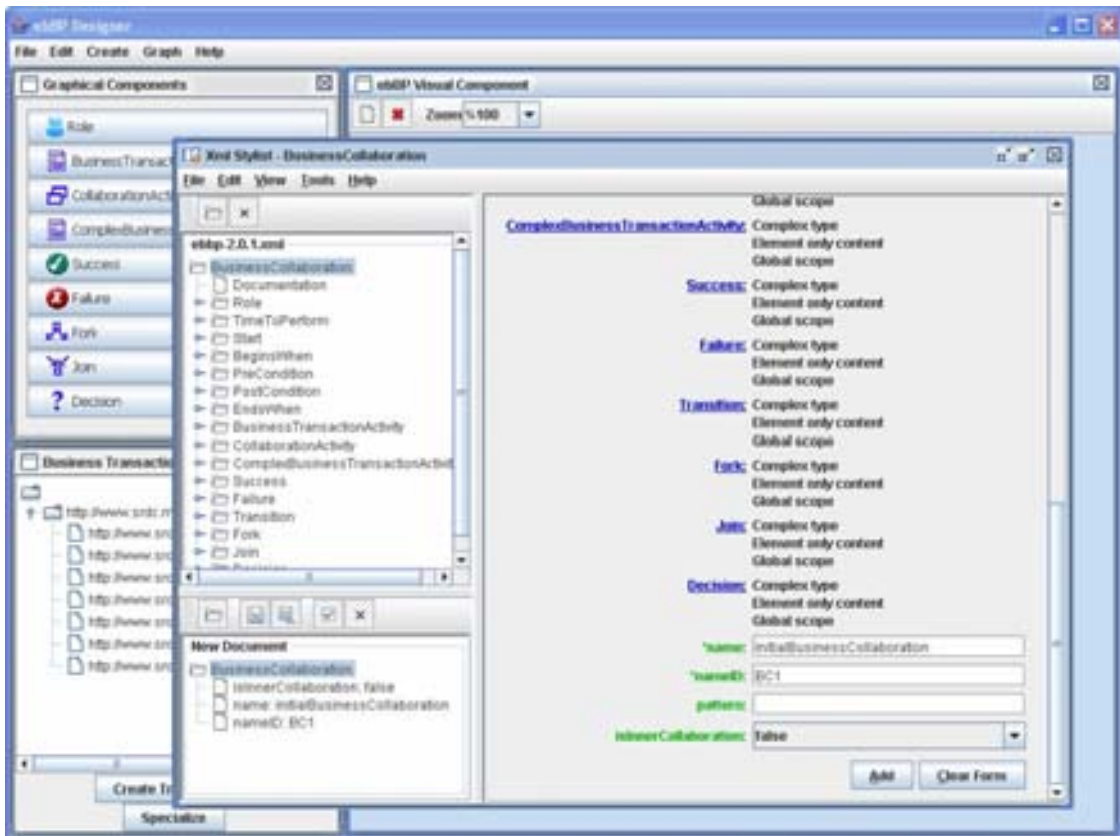


Figure 2 Initial View of the ebBP Editor

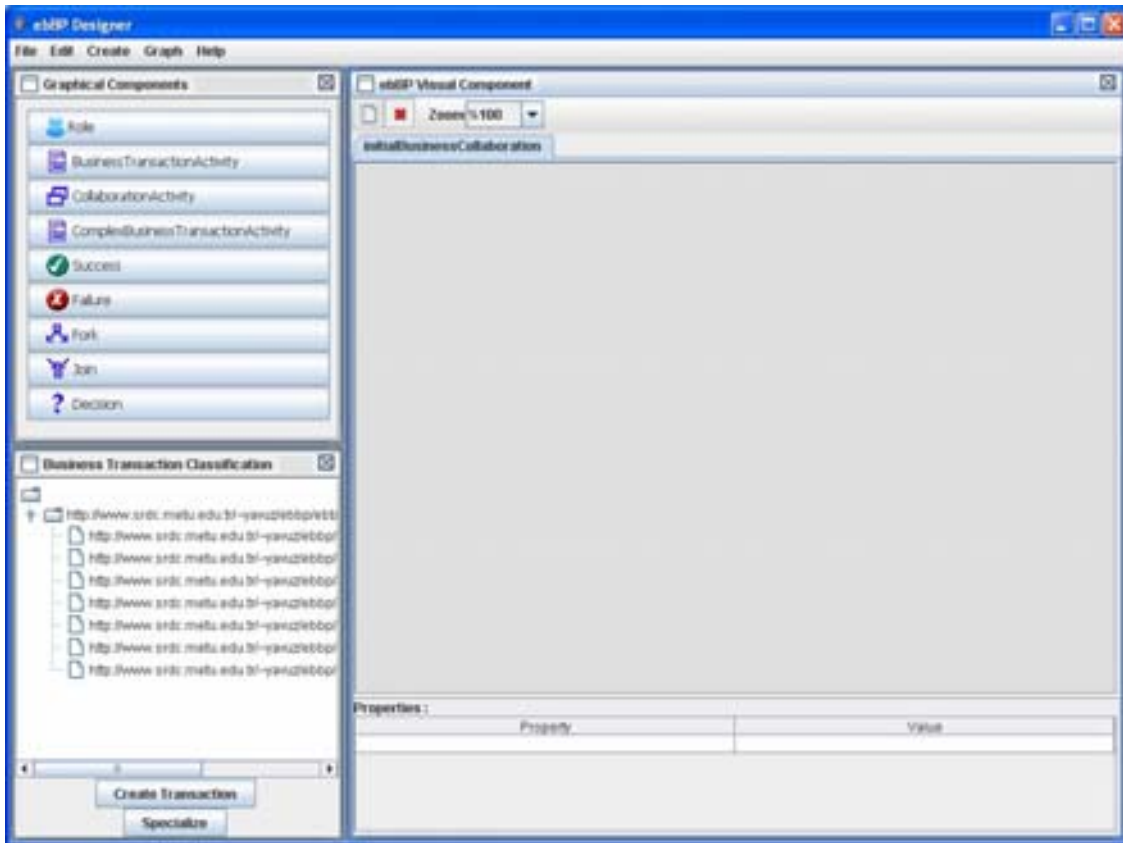


Figure 3 Initial View of the ebBP Editor

3.2 ebBP Editor components

The ebBP Editor has three internal windows representing its three components. This window design facilitates further extensions in the ebBP Editor. The current components are the “ebBP Visual Component” window, “Graphical Components” window, and “Business Transaction Classification” window (see Figure 3).

The ebBP Visual Component provides a visual representation of the Process Specification. However, instead of representing the whole Process Specification graphically, the Business Collaborations are presented in separate tabs (see Figure 4).

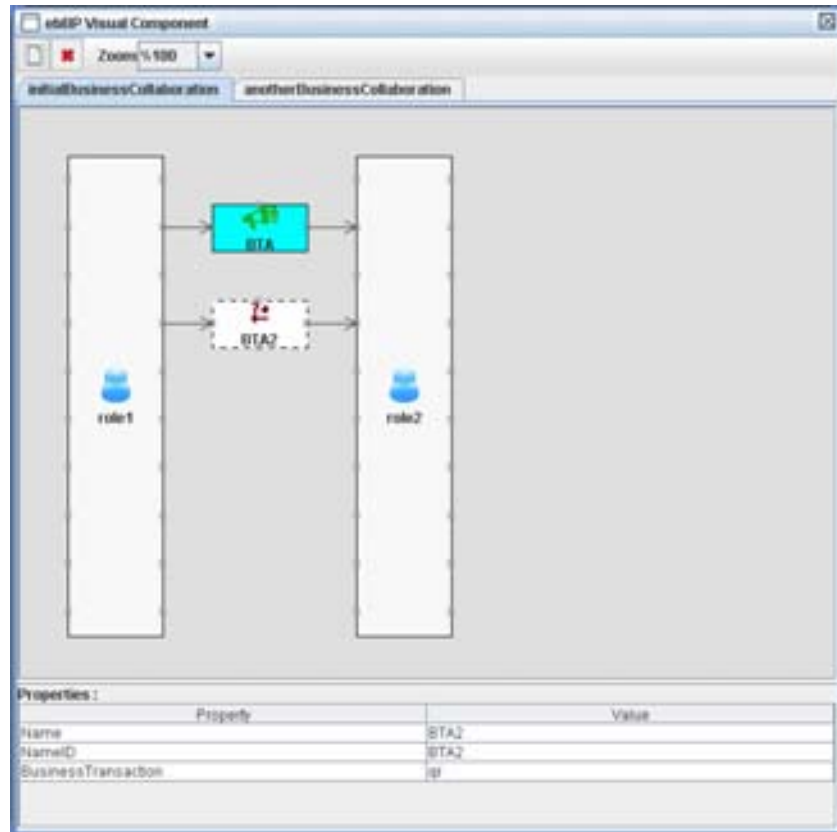


Figure 4 ebBP Visual Component

At the top-left there are “new Business Collaboration” and “remove Business Collaboration” icons. The ebBP Visual Component has two tabs: one for the initialBusinessCollaboration, and one for the anotherBusinessCollaboration. In addition, at the bottom of the window, the properties of the selected item can be seen.

The Graphical Components Panel displays the constructs that can be created within a Business Collaboration. These constructs are namely: Role, Business Transaction Activity, Collaboration Activity, Complex Business Transaction Activity, Success, Failure, Fork, Join, and Decision (see Figure 5). Dragging and dropping a button into the Business Collaboration tab, will create the corresponding ebBP construct.

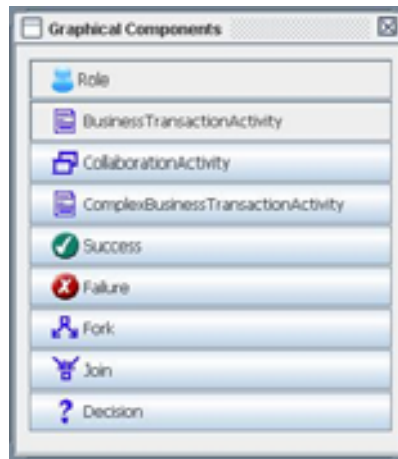


Figure 5 Graphical Components

Finally, the third component is the “Business Transaction Classification” Window (see Figure 6). This component provides a semantic infrastructure for the Business Transactions. The window presents a dynamic ontology representing the classification of Business Transactions, and lets the user specialize any of the Business Transactions or a specialized Business Transaction. The Specialized Business Transactions are then annotated in the ebXML registry.

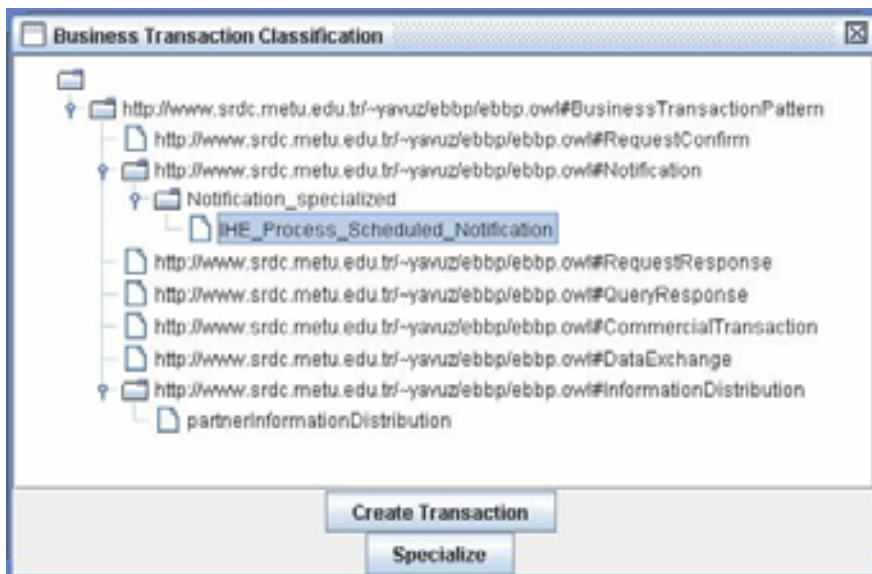


Figure 6 Business Transaction Classification

3.3 XMLStylist - XSD Editor

3.3.1 Abstract

Xml Stylist is a tool that generates XML instance documents from a given XML Schema by providing a graphical user interface. Xml Stylist parses the XSD file and dynamically creates the graphical user interface of the corresponding elements defined in the XSD. The user can then fill in the necessary fields by providing required information. The editor also provides the user with information about necessity, cardinality and optionality of the fields. Once the

user has filled in the information through the GUI, a validation check is made, and the information is serialized into XML.

3.3.2 Usage Areas

Xml Stylist has two major usages. First of all, its main usage is to create new XML instance documents that confront to a XSD document. Secondly, it can be used to edit existing XML documents. It may also be used to validate XML documents against an XML Schema file.

3.3.3 General View

The main screen of the Xml Stylist has 3 parts. At the top left part of the window, XSD file is visualized as a tree object. Below this, dynamically created XML file is shown. Right part of the window contains the selected XSD element details as a form. Figure 7 shows the general view of Xml Stylist.

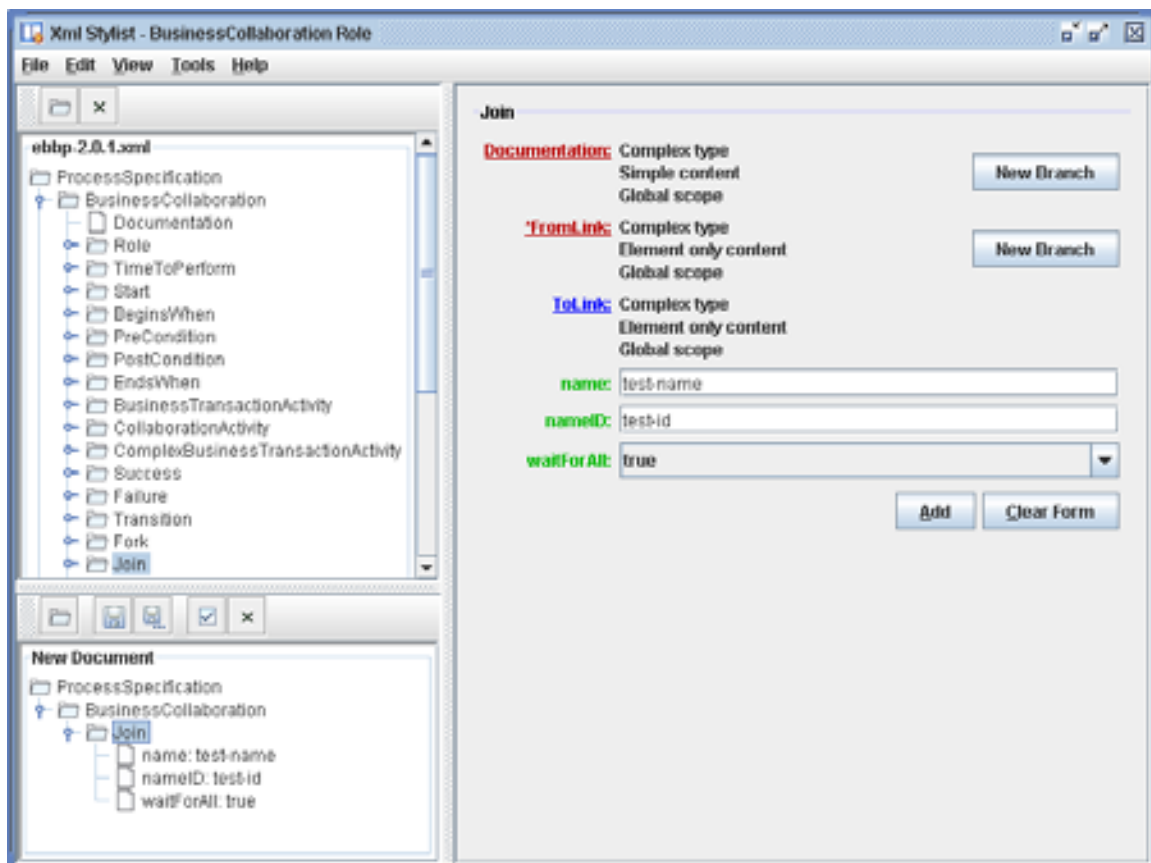


Figure 7 Xml Stylist - General View

3.3.4 General Usage

First of all, Xml Stylist forms the XSD tree by parsing the XSD file. When the user loads a XSD file, Xml Stylist checks if it has a single global element. If it contains multiple global elements, Xml Stylist shows the whole global elements and their children as a tree as shown in Figure 8. The user must specify the root element of the XML instance by selecting the appropriate element.

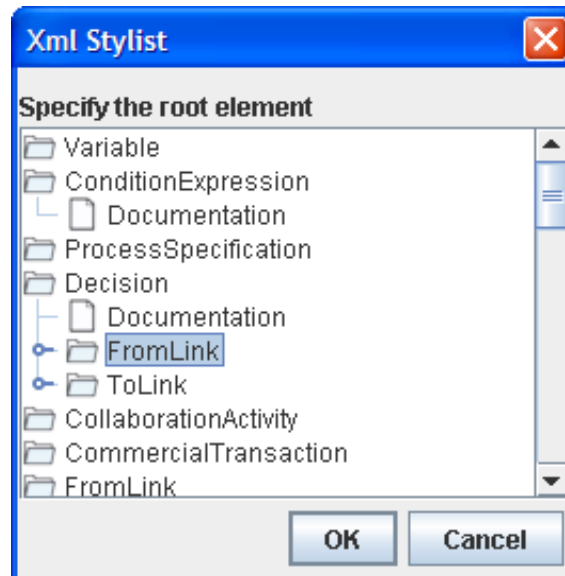


Figure 8 Xml Stylist - Root Select Dialog

After that, the user can select the elements of the XSD file by clicking the XSD tree. A form, corresponding to the selected element is created dynamically at the right side of the window as shown in Figure 9.

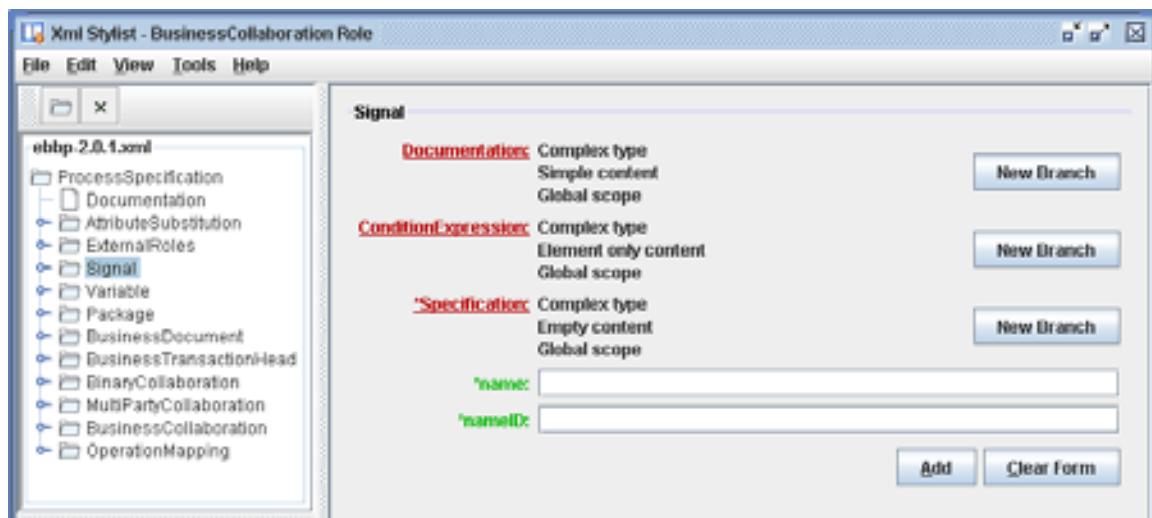


Figure 9 Xml Stylist - Form

This form is created as follows:

- Simple elements are shown as a text field, if their maximum occurrences are 1, or as a combo box, if their maximum occurrences are greater than one.
- Complex elements are shown as a link to their form. When you click their name on the form, it selects this element from the XSD tree and its form is displayed in the form. If this elements maximum occurrence is greater than 1, a new button appears next to the link named “New Branch” which adds a new element to the XML document.
- Attributes are also shown as a text field.

Field labels give valuable information. As Figure 10 shows, if you hold your mouse over a label, a tool tip that displays namespace, scope and cardinality information appears. If the field is a required field, a star (*) appears before its label and the label is written in bold. Fields also have tool tips that show the type of that field.

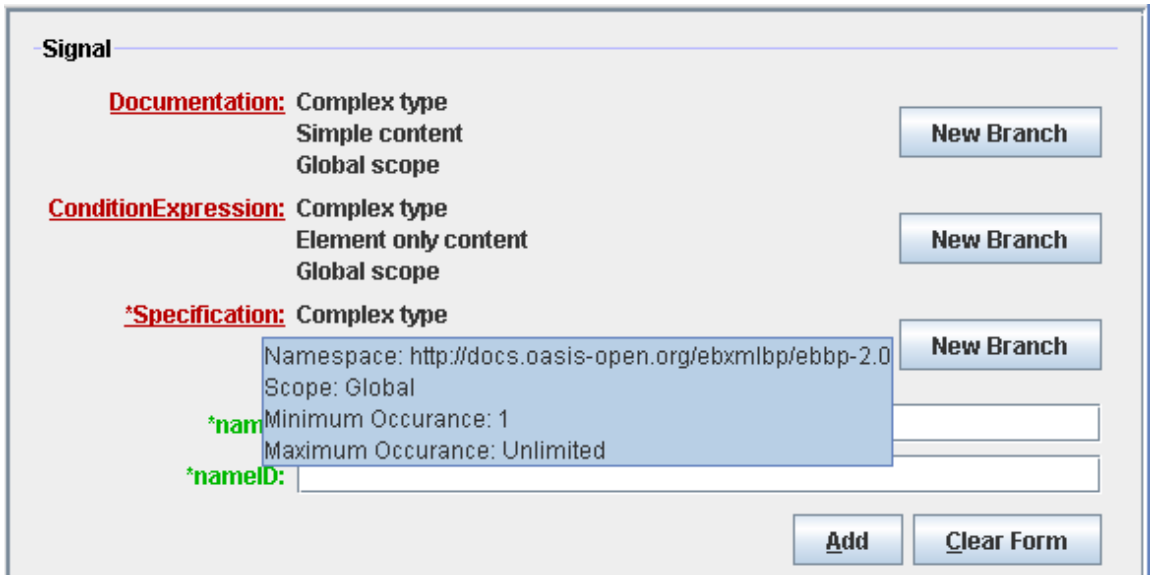


Figure 10 Xml Stylist – Tool tips

3.3.5 Validation

Validation can be performed any time by clicking the validate toolbar button, clicking the Tools – Validate menu item in the menu bar or pressing the F11 button. This process is shown in Figure 11.

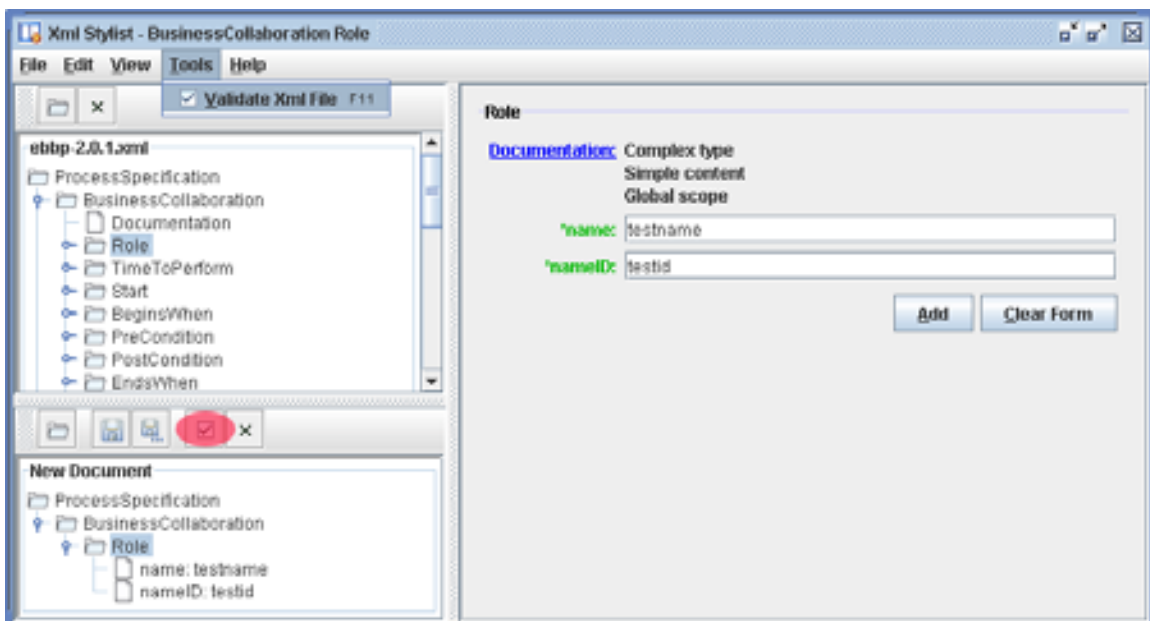


Figure 11 Xml Stylist – Validation

3.3.6 Limitations

Xml Stylist uses text fields for simple type elements that have at most one occurrence and combo boxes for simple type elements that have more than one occurrence. In order to delete a simple type element that is visualized as a text field is easy. Just delete the text in the text field and click the Add button. But deleting from a combo box is not possible in this version.

Xml Stylist also can not support the whole XML Schema specifications right now. No type checking is made in text fields / combo boxes.

3.4 Creating a Business Collaboration

A Business Collaboration can be created by clicking on the new Business Collaboration icon on the “ebBP Visual Component” (see Figure 12). This will create a new Tab in the ebBP Visual Component Window.



Figure 12 New Business Collaboration

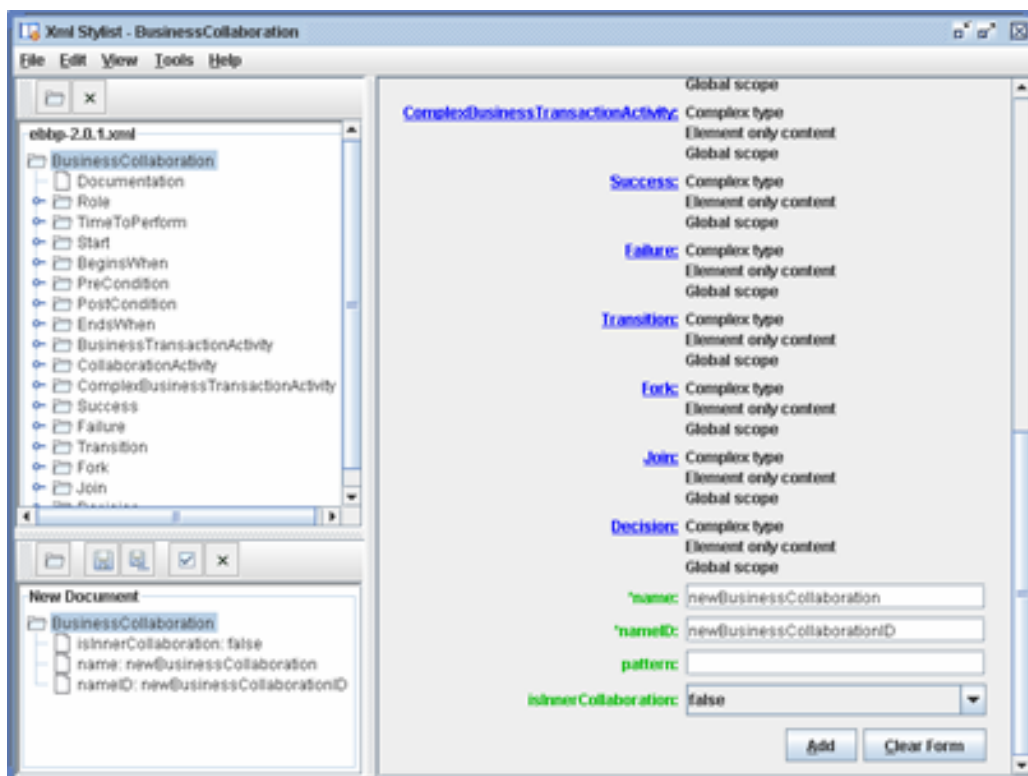


Figure 13 New Business Collaboration - XSD Editor

The XSD Editor will come up, so that the user can provide details about the Business Collaboration as in **Error! Reference source not found.**. After the user provides the information and closes the XSD Editor, a new Business Collaboration tab will be created in the ebBP Visual Component (see Figure 14).

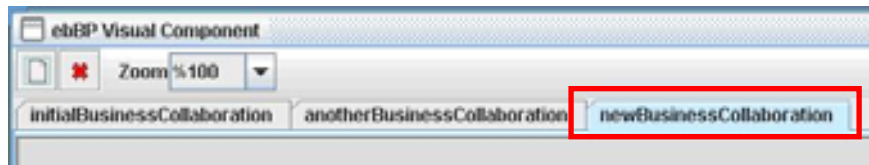


Figure 14 New Business Collaboration Tab

3.5 Removing a Business Collaboration

A Business Collaboration can be removed by clicking on the remove Business Collaboration icon on the “ebBP Visual Component” (see Figure 15). After removing the Business Collaboration, the tab representing this Business Collaboration will disappear.



Figure 15 Remove Business Collaboration

3.6 Creating a Role

A Role can be created by dragging and dropping the role button on the “Graphical Components” window into the ebBP Visual Component. The role will be created part of the Business Collaboration within the Tab selected.

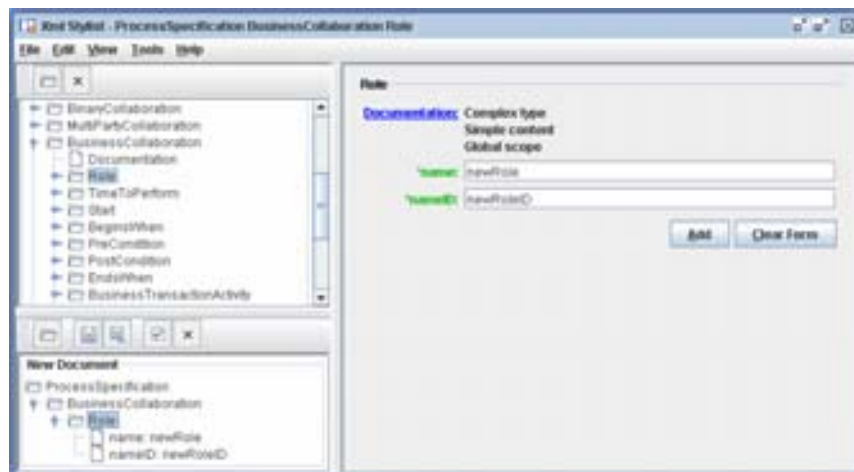


Figure 16 New Role

First, the XSD Editor will come up. The user is supposed to fill in the Role details (see Figure 16). After providing the information and closing the XSD Editor, the new Role will be placed into the Business Collaboration (see Figure 17).

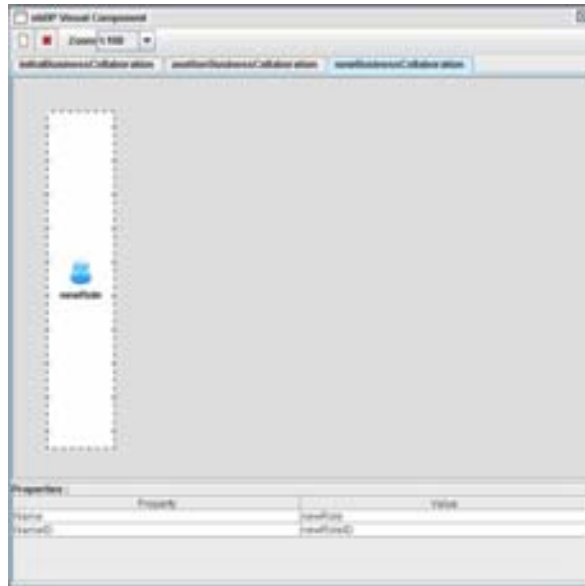


Figure 17 New Role Representation

3.7 Creating a Business Transaction Activity

A Business Transaction Activity can be created by dragging and dropping the Business Transaction Activity button on the “Graphical Components” window into the ebBP Visual Component. The Business Transaction Activity will be created part of the Business Collaboration within the Tab selected.

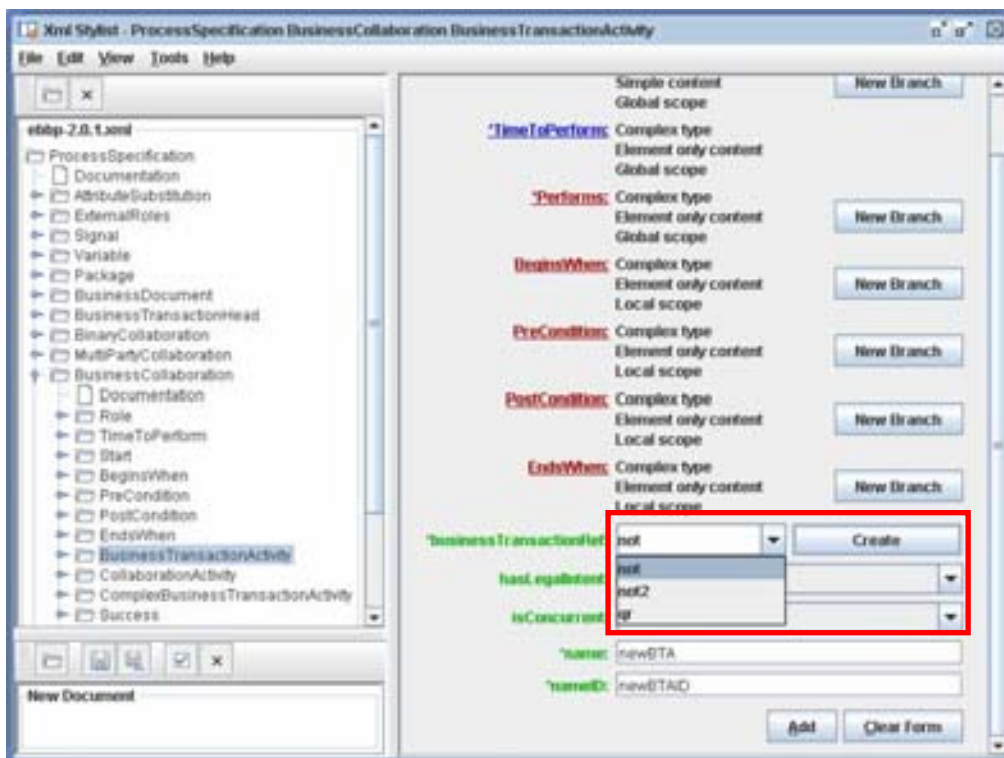


Figure 18 New Business Transaction Activity

As shown in Figure 18 the user should first select the businessTransactionRef field. The combo box displays all Business Transactions (Notification, Commercial Transaction, Query Response, Request Response, Request Confirm, Information Distribution, and Data Exchange) previously created. The user may choose one of these, or decide to create a new Business Transaction.



Figure 19 Creating a New Business Transaction

If the user decides to create a new Business Transaction and presses the Create button, a dialog will come up presenting the possible Business Transactions. The user can create any of these Business Transactions. The XSD Editor will come up to create the new Business Transaction. The user should not forget to create the Requesting and Responding roles of the Business Transactions. These are later used in associating the Roles in the Business Collaboration with the roles of the Business Transaction.

Upon closing the XSD Editor, the Business Transaction will be saved, and accessible from the combo box. (Note: At first the item may not appear in the combo box: either refresh the form, reselect the BusinessTransactionActivity node at the left (see Figure 20); or directly type in the nameID of the newly created Business Transaction into the combo box)

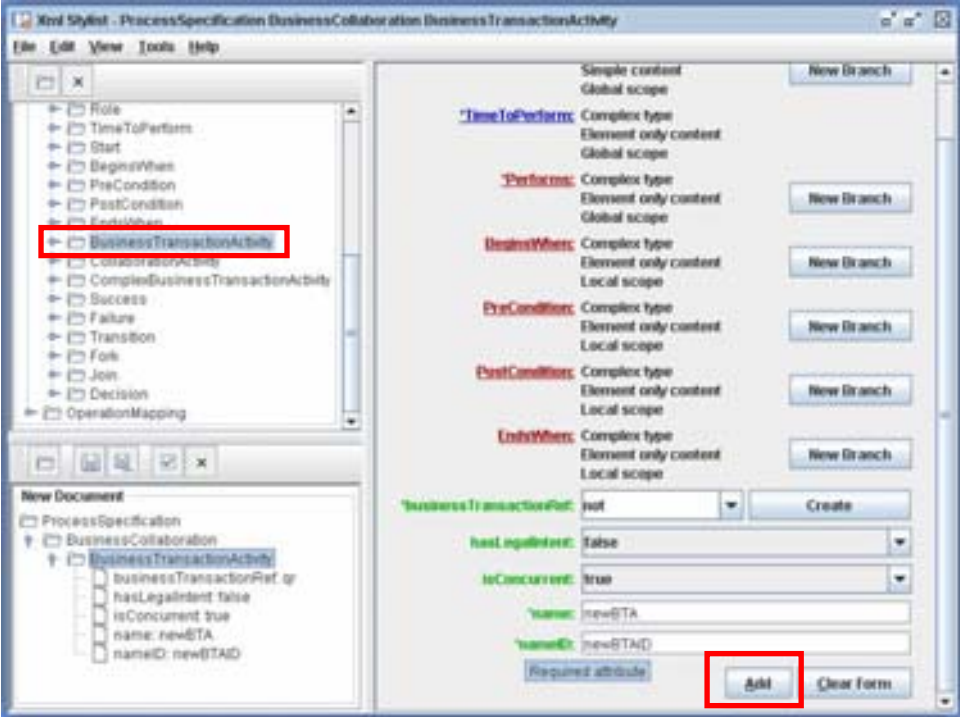


Figure 20 New Business Transaction Activity

After filling in the details of the Business Transaction Activity and pressing the Add button (see Figure 20), the user should close the XSD Editor.

In Figure 21 the representation of the newly created Business Transaction Activities can be seen. There are two Business Transaction Activities to explain the representation in detail. Note that the first one (newBTA) is in blue, whereas the second one (BTA2) is white. This difference is due to the fact that the first one represents the Start construct as well. The user cannot add a Start construct by default, but the first Business Transaction Activity is processed to be the Start. This can be changed by the user by replacing the Business Transaction Activities by dragging.

Another difference is in the icons. Note that both Business Transaction Activities have different icons. This is because of the Business Transaction types they reference. The first one (newBTA) references a Query-Response Business Transaction, whereas the other references a Notification. Other types of Business Transactions also result in different icons.

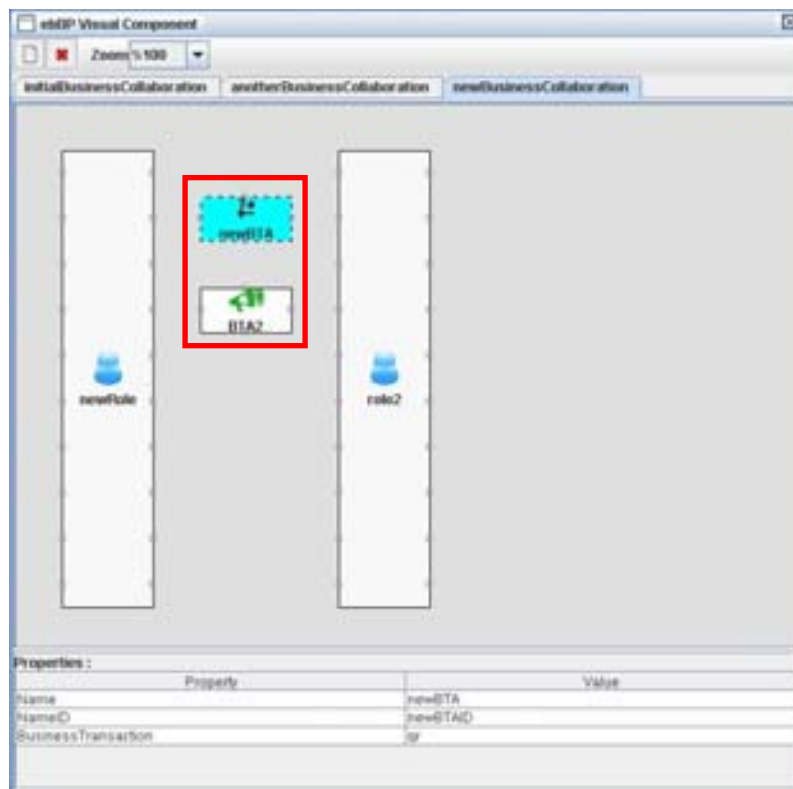


Figure 21 Business Transaction Activity Representation

Now that we added the Business Transaction Activity, the roles should be associated with it. By dragging the ports of the Roles to the Business Transaction, the requesting Role can be set. Similarly, the requesting role can be set by dragging a port from the Business Transaction Activity to the Role (see Figure 22).



Figure 22 Associating the Roles

3.8 Creating a Collaboration Activity

A Collaboration Activity can be created by dragging and dropping the Collaboration Activity button on the “Graphical Components” window into the ebBP Visual Component. The Collaboration Activity will be created part of the Business Collaboration within the Tab selected.

The Collaboration Activity references a Business Collaboration. When a Collaboration Activity is created, the XSD Editor presents the user available Business Collaborations. The user may select one of the available Business Collaborations, or decide to create a new one. When a new one is created (see section 3.4) a new tab is added accordingly.

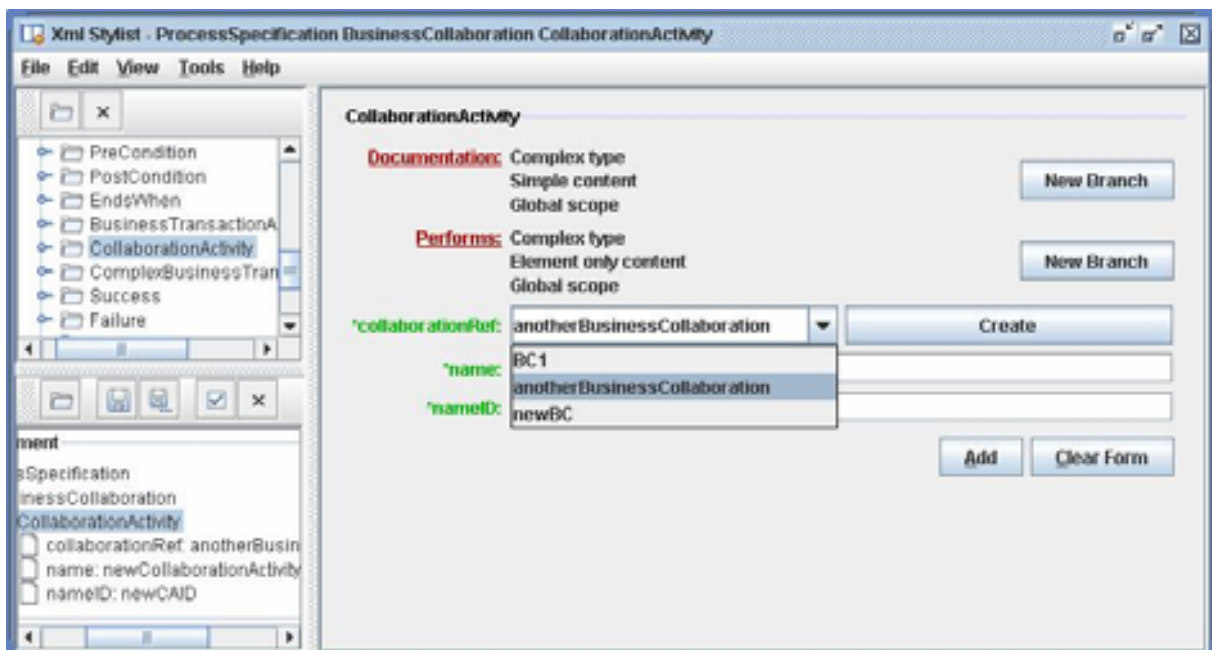


Figure 23 New Collaboration Activity

After closing the XSD Editor, the collaboration will be placed into the Business Collaboration as in Figure 24.

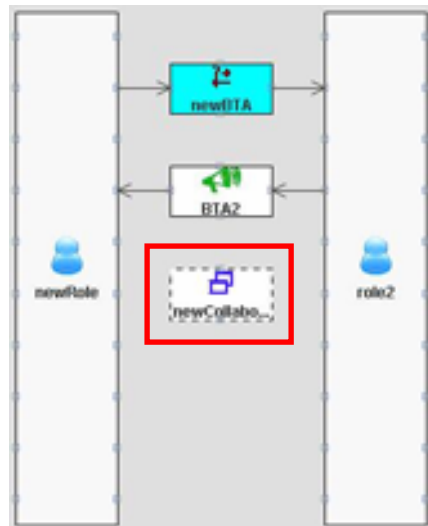


Figure 24 Business Collaboration Representation

3.9 ***Creating a Complex Business Transaction Activity***

A Complex Business Transaction Activity can be created by dragging and dropping the Complex Business Transaction Activity button on the “Graphical Components” window into the ebBP Visual Component. The Complex Business Transaction Activity will be created part of the Business Collaboration within the Tab selected.

3.10 ***Creating a Success***

A Success can be created by dragging and dropping the Success button on the “Graphical Components” window into the ebBP Visual Component. The Success will be created part of the Business Collaboration within the Tab selected.

3.11 ***Creating a Failure***

A Failure can be created by dragging and dropping the Failure button on the “Graphical Components” window into the ebBP Visual Component. The Failure will be created part of the Business Collaboration within the Tab selected.

3.12 ***Creating a Fork***

A Fork can be created by dragging and dropping the Fork button on the “Graphical Components” window into the ebBP Visual Component. The Fork will be created part of the Business Collaboration within the Tab selected.

3.13 ***Creating a Join***

A Join can be created by dragging and dropping the Join button on the “Graphical Components” window into the ebBP Visual Component. The Join will be created part of the Business Collaboration within the Tab selected.

3.14 Creating a Decision

A Decision can be created by dragging and dropping the Decision button on the “Graphical Components” window into the ebBP Visual Component (see Figure 25). The Decision will be created part of the Business Collaboration within the Tab selected.

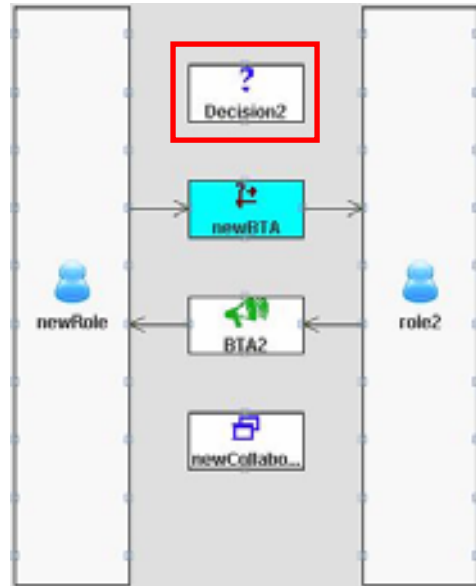


Figure 25 New Decision

After creating the Decision, the user can create the FromLink and ToLink constructs. Dragging from a Business Transaction Activity port to a Decision port will create a FromLink. On the other hand, a ToLink will be created dragging vice versa (see Figure 26).

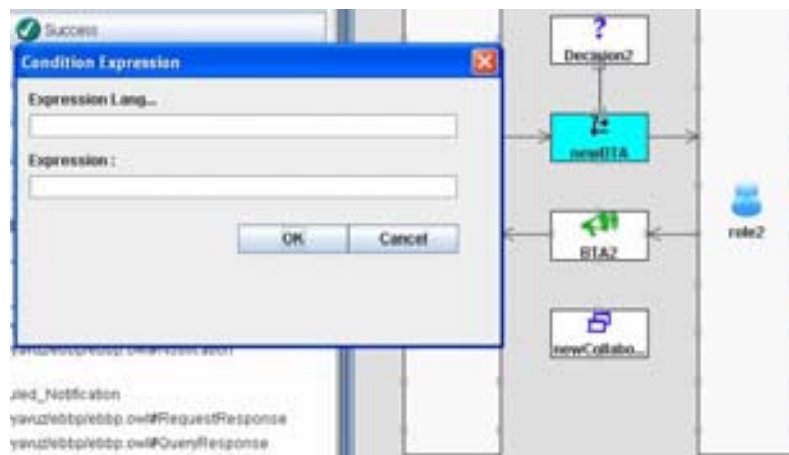


Figure 26 Decision Connections (ToLink, FromLink)

When a ToLink is created the user will be prompted to provide the condition expression. Multiple ToLink's can be created.

3.15 Specializing a Business Transaction

Business Transactions can be specialized by selecting a node from the ontology, which represents a Business Transaction, and then pressing the Specialize button (see Figure 27). The XSD Editor will come up with specialized fields, if any. When the non-specialized (raw) Business Transactions are selected for specialization, the fields will all be empty. The user can fill in any detail regarding the Business Transaction.

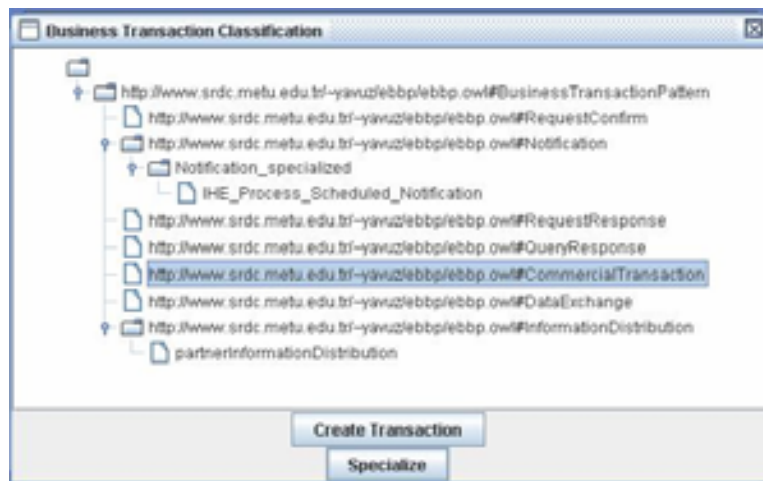


Figure 27 Business Transaction Specialization

When finished (please remember to press the Add button to save your entries), the user should close the XSD Editor. Upon closing the XSD Editor, a Specialization window is opened (see Figure 28). This window specifies the annotation for the newly specialized Business Transaction.

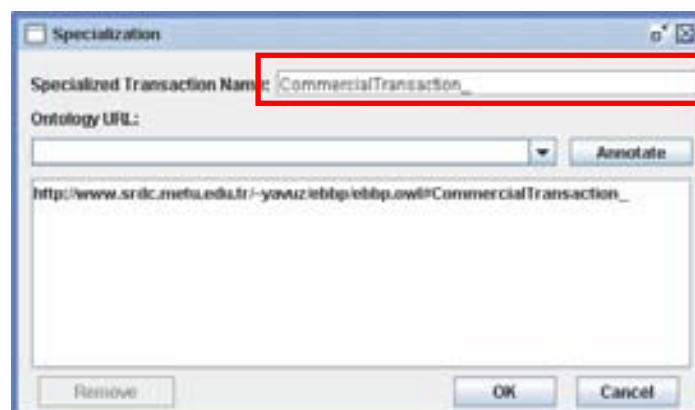


Figure 28 Specialization Annotation

The user should provide a name for the newly specialized Business Transaction. With this name a new subclass will be created in the dynamic ontology, and the transaction will be annotated with the Classification Node in the ebXML registry, that will be created dynamically. Note that, the user cannot remove this annotation from the list.

However, the user may annotate the transaction with a node within a domain-specific ontology as well, by entering the ontology URL into the field, and then pressing the Annotate

button (see Figure 29). This will bring up the Annotation Window. The user can explore the ontology and select a node from the ontology, either using the tree representation or the alphabetically ordered list. Both representations have also a filter associated that filters the nodes beginning with the entered text (see Figure 29). When the user presses the Annotate button on this window (see Figure 29), the selected node is added to the annotation list. The user may remove this node later, after selecting the node and pressing the remove button.

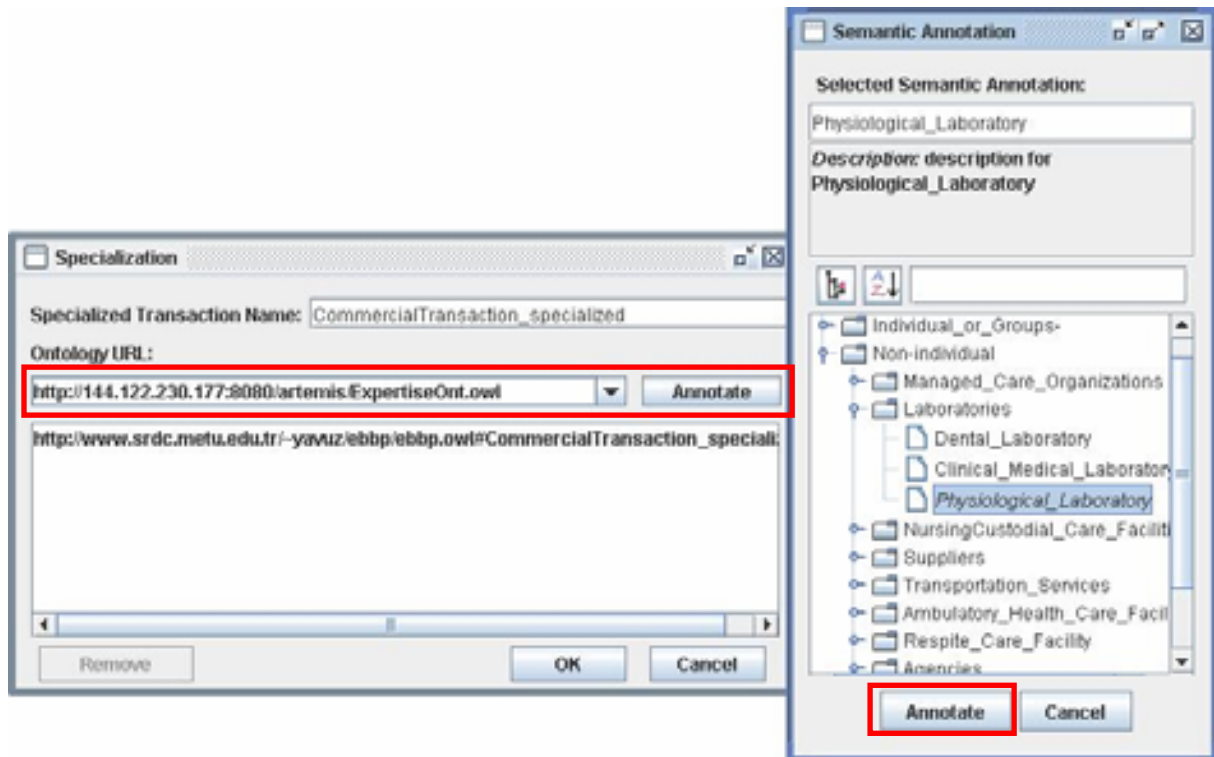


Figure 29 Domain-specific Annotation

After adding all nodes for annotation, the user should press the OK button to create the specialized Business Transaction (a template), and annotate it with the classification nodes in the ebXML registry (see Figure 30). If Cancel is pressed no action will be taken.

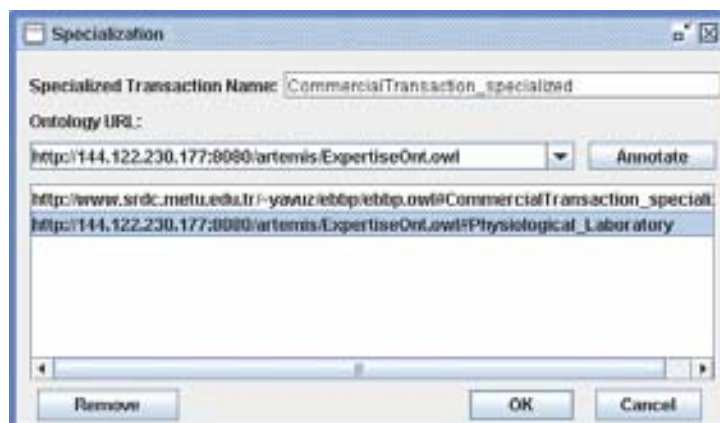


Figure 30 Annotations

The newly specialized Business Transaction can be seen in Figure 31.

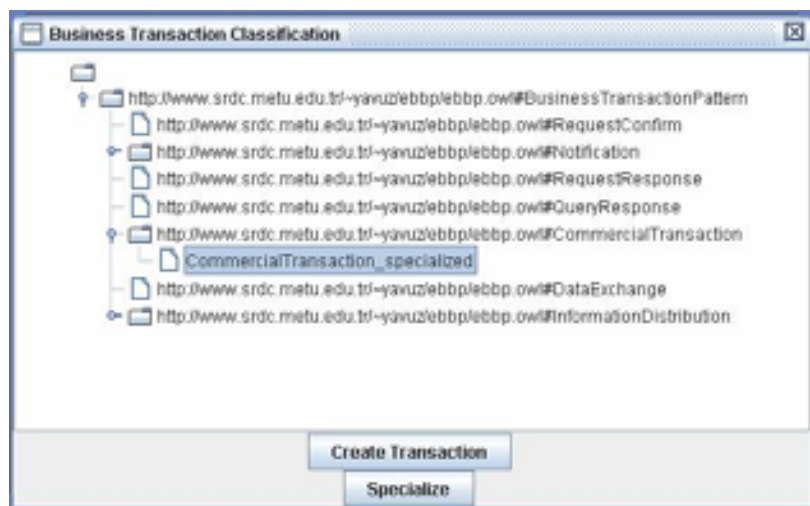


Figure 31 Specialized Commercial Transaction

To exemplify this specialization and annotation process, suppose the user wishes to specialize a Notification as a “Procedure Scheduled” transaction of IHE. The user may set the BusinessDocument in the DocumentEnvelope of the RequestingBusinessActivity to be sent by the requesting role which is part of the Notification. After setting the domain specific fields, the user annotates the “Procedure Scheduled Notification”. The user may name this notification “IHE Procedure Scheduled Notification” and this procedure may be specialized under the “Scheduled Notification” node which could be a subclass of the “Notification” transaction. In this case, the ebBP (template) definition corresponding to “IHE Procedure Scheduled Notification” will first be annotated with `http://www.srdc.metu.edu.tr/ebbp.owl#IHE Procedure Scheduled Notification`, which is created as a subclass of `http://www.srdc.metu.edu.tr/ebbp.owl#Scheduled Notification`. In addition, the user may specify that this notification should also be annotated with `http://www.srdc.metu.edu.tr/ihe.owl#Procedure Scheduled`. Then, the ebBP editor, will place the ebBP (template) definition into the ebXML repository and annotate it with the classification nodes specified by the user.

3.16 Creating a Business Transaction

A Business Transaction can be created with two approaches: top-down, bottom-up. This is mainly due to the fact that the Business Transactions are not directly represented in the ebBP Visual Component. The Business Transactions should be part of a Business Transaction Activity. Therefore, the top-down approach is as follows: When adding a Business Transaction Activity (see Section), the user can provide the businessTransactionRef reference. This field is presented as a combo box, which presents the user with all Business Transactions created so far (as well as those in other Process Specifications with XInclude, see Section 3.20). The user can select (reuse) a previously created Business Transaction, or decide to create a new one, by pressing the Create button. When the user presses the Create button, a new XSD Editor window will be opened. The user can create a new Business Transaction with this instance of the XSD Editor.

The bottom-up approach is explained in the next section.

3.17 *Creating a Specialized Business Transaction*

The bottom-up approach is as follows: The user may drag and drop a Business Transaction or a specialized Business Transaction from the Business Transaction Classification Window. The user will be presented with an instance of the XSD Editor. The XSD editor will come up empty for the top level Business Transactions (non-specialized), or some fields pre-filled for the specialized Business Transactions. The user is supposed to fill in at least the rest of the required fields and close the XSD Editor window. Upon closing the XSD Editor, a default Business Transaction Activity will be created, that references the newly created Business Transaction.

3.18 *Editing the Process Specification*

The user may edit the Process Specification from the Edit→Process Specification menu. The XSD Editor will come up with the Process Specification created so far. The user can make changes which will be saved after the XSD Editor is closed.

3.19 *Saving the Process Specification*

The Process Specification can be saved from the File→Save menu. A Save Dialog is opened to determine the File location and File name. If the Process Specification was previously saved it is directly overwritten.

3.20 *Importing another Process Specification (XInclude)*

The ebBP Editor also supports XInclude. The user can Import a file through the File→Import menu. An open dialog will come up to determine the file to be imported. After importing the file, XInclude structures can be used. However, this is not reflected to the user. XInclude operations are possible only where fields are represented with combo boxes. Selecting a value existing in another file will result in the addition of the XInclude when the Process Specification is saved.

4 Conclusion

This document provides a basic User Guide presenting the functionalities and capabilities of the ebBP Editor. Although the implementation provides a user interface for creating the Process Specification complying with the latest ebBP Specification, there are still some functionalities that need to be implemented. The editor also reflects some development based on ongoing research. In this respect, later versions may have extended functionalities, or changes in the implementation.