

# Laboratorio: Creación de un flujo BPEL con WID

## Objetivos

El objetivo de este laboratorio es enseñar a los alumnos a crear un nuevo proyecto de servicios con un flujo BPEL básico usando WebSphere Integration Developer v.6.01 (WID).

En este laboratorio crearemos un flujo básico que simplemente regresa un mensaje de texto ("Esta cadena es corta" o "Esta cadena es larga" dependiendo de la longitud del mensaje de entrada invocando para ello un servicio web que calcule la longitud de la cadena original.

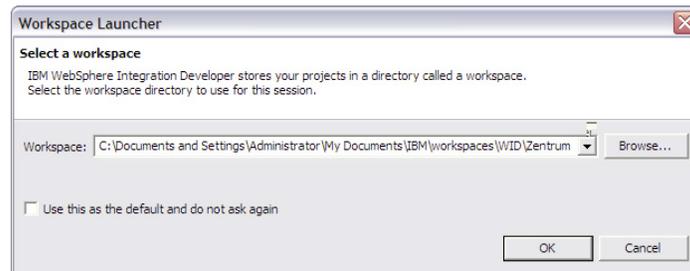
## Prerrequisitos

WID debe estar instalado en su equipo.

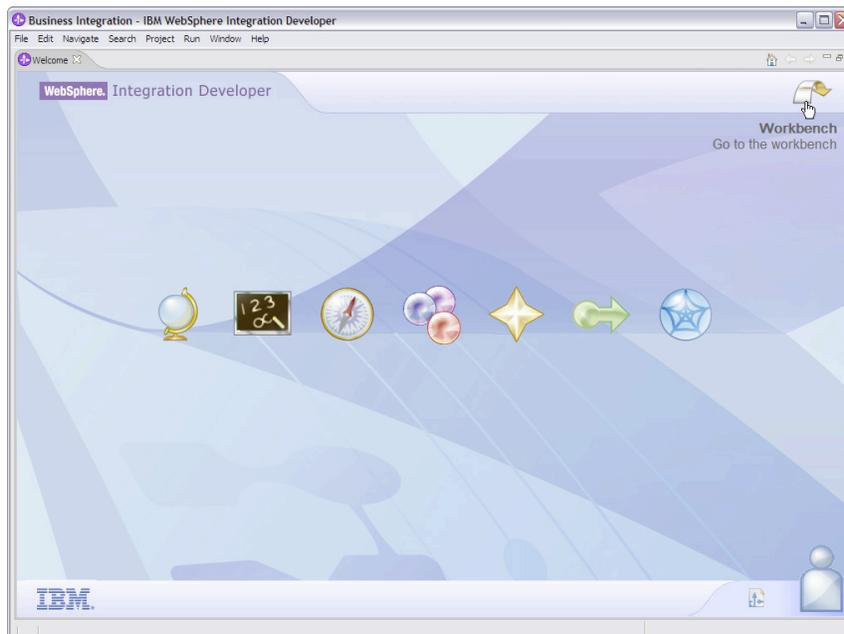
## Instrucciones

Inicie el entorno de desarrollo pulsando el siguiente icono  en la barra de tareas.

A continuación aparece una pantalla en la que se solicita la ubicación de un directorio en el que se almacenarán todos los archivos del proyecto (workspace).



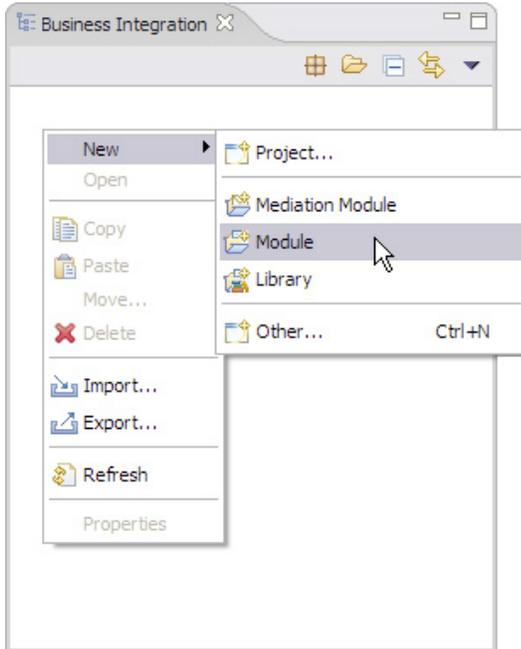
Una vez completado este paso, el entorno de desarrollo arranca y aparece la siguiente pantalla



Cierre la pestaña de Bienvenida haciendo click sobre la gran flecha situada en la esquina superior derecha de la pantalla para acceder al workbench.

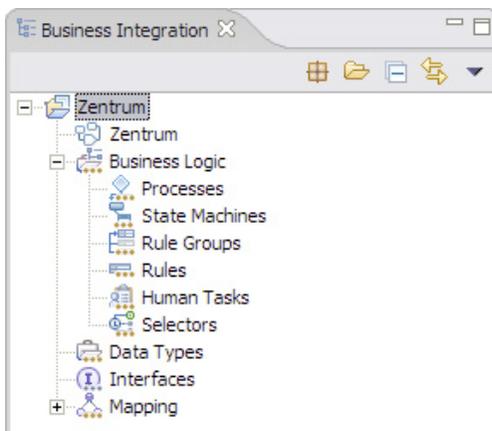
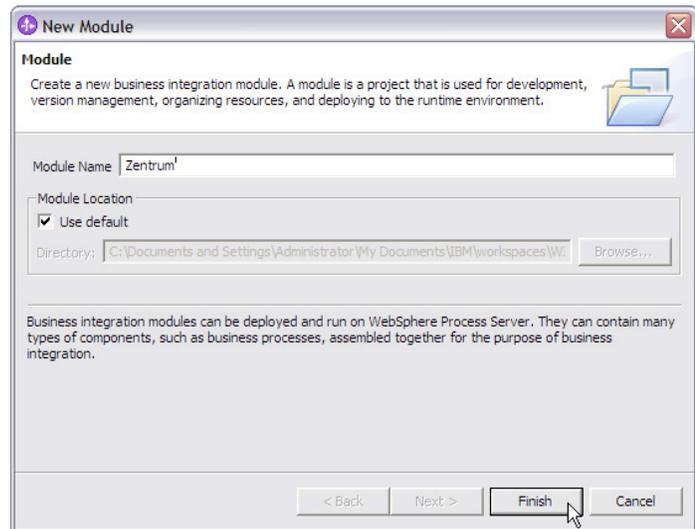
Ahora ya estamos listos para empezar a trabajar en nuestro proyecto.

WID es un producto que tiene muchos componentes en común con Rational Application Developer por lo que permite realizar muchas de las mismas operaciones soportadas por este producto. Por ejemplo, podemos utilizar WID para crear aplicaciones de web dinámicas o Enterprise Java Beans. Sin embargo, una de las principales características que diferencia a WID es que este producto permite crear proyectos de integración. Este tipo de proyectos, a pesar de que generan de manera automática código J2EE estándar, puede ser creado por personas que no tengan conocimientos de Java.



Para crear un nuevo proyecto de servicios, haga click con el botón derecho presionado (right-click) sobre la ventana de Business Integration, tal y como se muestra en la ilustración de la izquierda. Seleccione New >> Module.

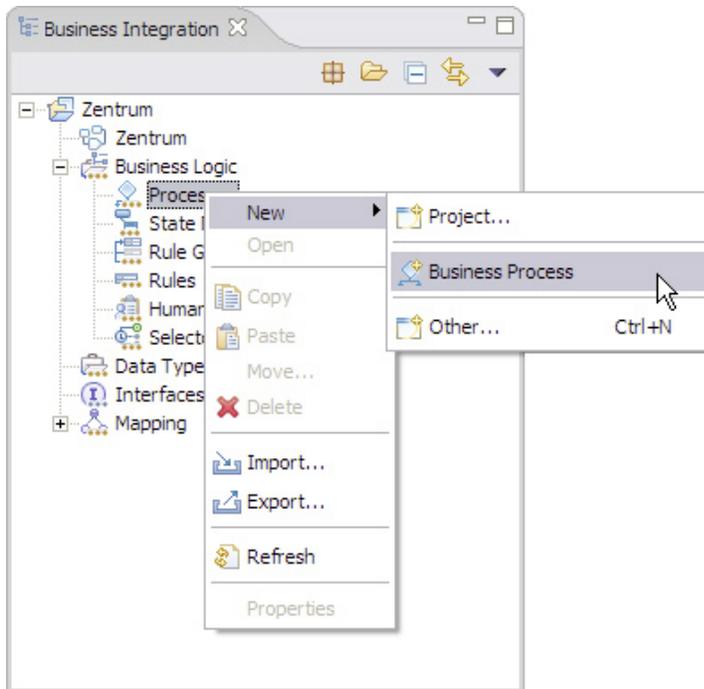
Cuando aparezca la pantalla que se muestra más adelante, elija un nombre para su proyecto y pulse Finish. Si hubiera pulsado el botón de "Next >" hubiera podido configurar las librerías de Java que su proyecto pudiera necesitar. Sin embargo, en este ejemplo, no dependemos de librerías externas y de todas maneras es algo que se puede configurar en cualquier momento.



Ahora que el proyecto ya está creado, podemos empezar a crear nuestro primer proceso BPEL. Esto se hace pulsando con el botón derecho del mouse (right-click) sobre el nombre del proyecto de servicios que acabamos de crear. tal y como se muestra en la ilustración que aparece a continuación.

De la misma manera en la que tuvimos que hacerlo al crear un nuevo proyecto, es necesario dar un nombre al nuevo proceso que vamos a crear.

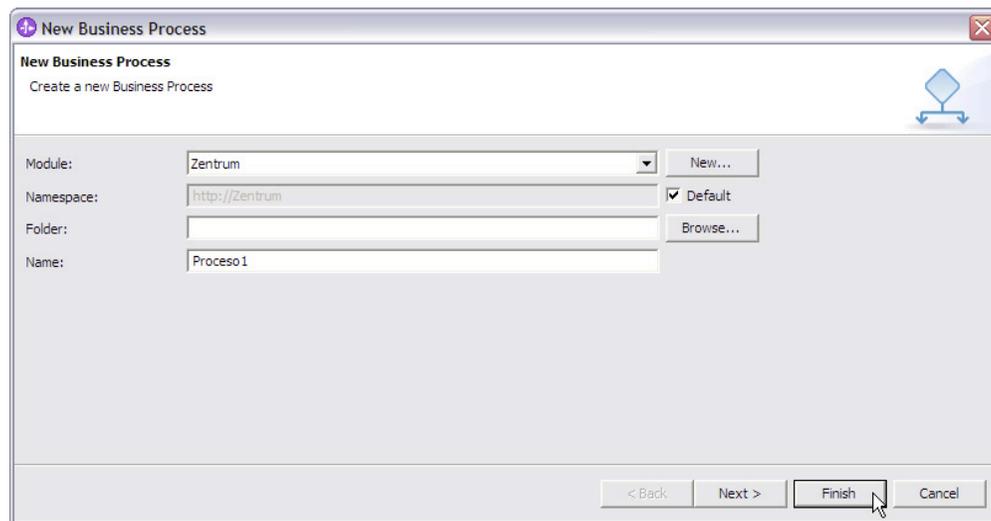
Cada uno de los componentes que se pueden crear dentro de un módulo en realidad es tan solo un archivo XML. Por ejemplo los procesos son archivos BPEL (Business Process Execution Language for Web Services), los cuales están basados en XML. Por lo tanto, necesitan estar definidos en un namespace propio para evitar conflictos con otros archivos XML que puedan tener el mismo nombre. Por defecto WID usa para los archivos que crea un namespace basado en el nombre del módulo y en este caso no existe ningún motivo para cambiarlo.



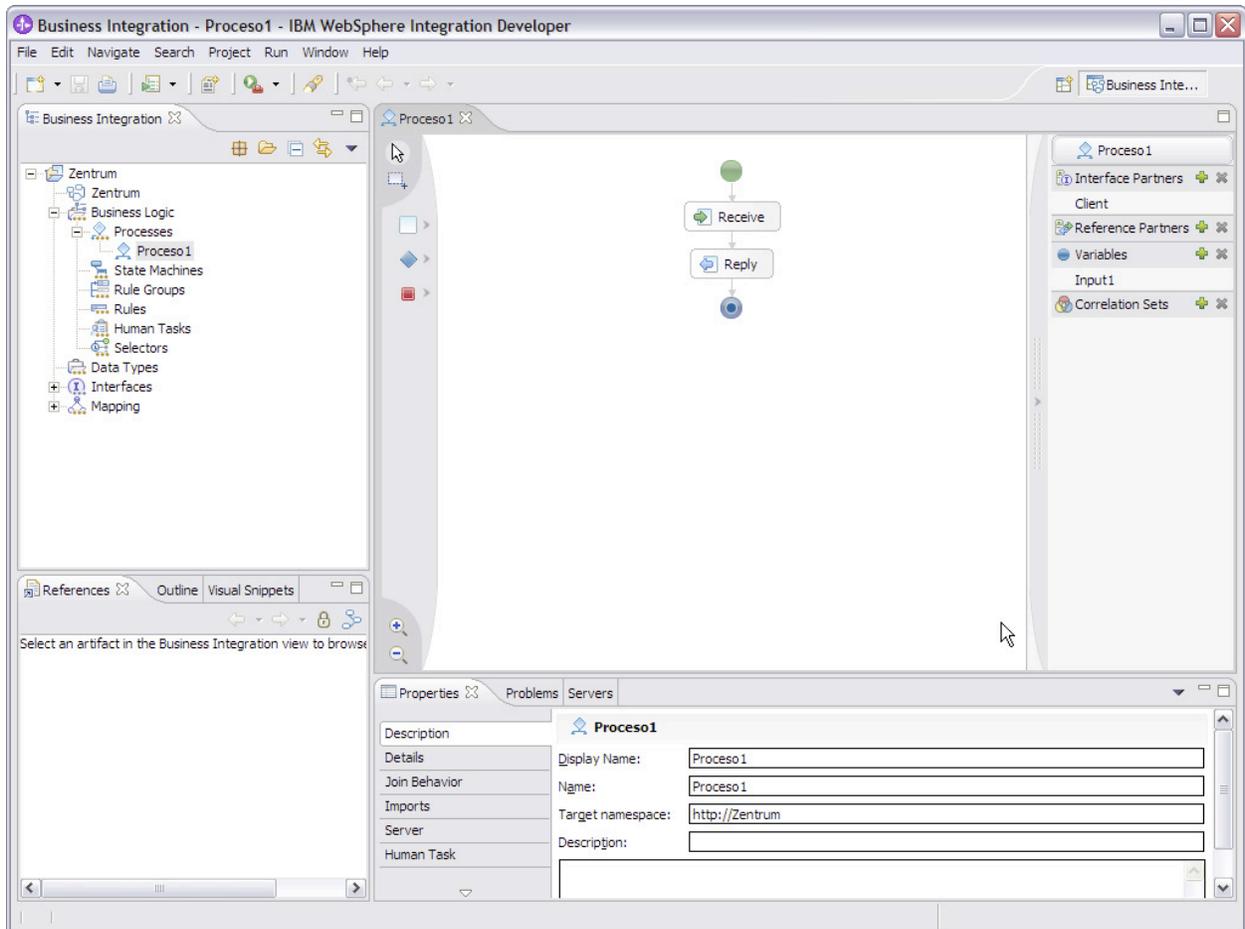
Por lo tanto, nuestro primer proceso al que llamaremos habilmente Proceso1 estará en el namespace `http://Zentrum`, tal y como se muestra a continuación.

BPEL es un estándar de la industria relativamente joven por lo que todavía tiene ciertas limitaciones. Por ejemplo no permite definir más que procesos totalmente automatizados sin intervención humana, algo que es muy necesario para modelar procesos complejos. Por esa razón IBM ha propuesto extender el estándar para soportar esta y otras características. WID permite ya incluir esta funcionalidad adicional en los procesos pero si el desarrollador decide hacerlo ya no podrá ejecutar el proceso en otros motores BPEL. WID permite bloquear el uso de las funciones no estándar. Para ello puede pulsar "Next >" dos veces y marcar la opción correspondiente. En este caso no es necesario ya que de todas maneras no vamos a usar la funcionalidad adicional.

Pulse "Finish" para completar la operación tal y como se muestra en la siguiente pantalla.



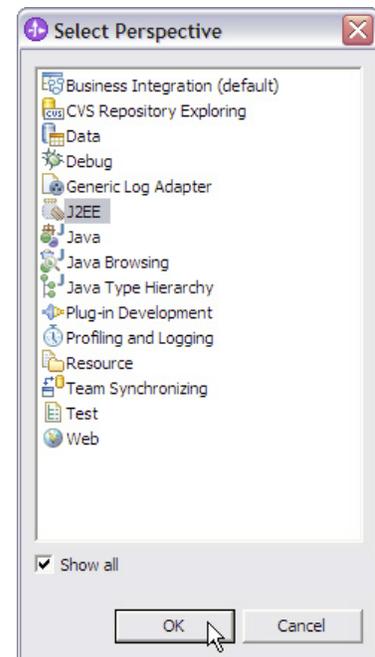
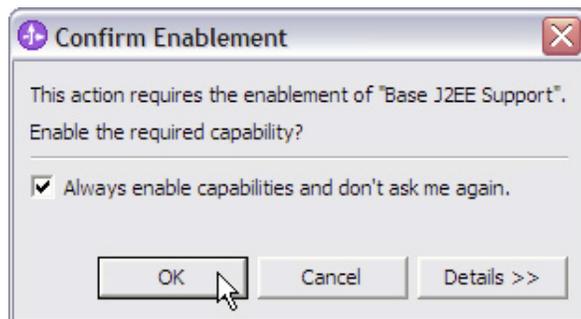
El entorno de desarrollo genera los archivos necesarios para definir un proceso básico sin acciones y que utiliza una variable llamada Input1 de tipo String tanto como parámetro de entrada como de salida. Al finalizar, este es el aspecto que muestra la herramienta de desarrollo.



Ahora vamos a crear el servicio web que calcula la longitud de una cadena de caracteres. Para ello vamos a necesitar cambiar a la perspectiva de J2EE.

Agregue la perspectiva de J2EE pulsando el ícono . Seleccione "Other...". Marque la opción "Show all" y elija la perspectiva de "J2EE". Pulse el botón "OK" para continuar.

Cuando el producto le solicite confirmar si desea activar la funcionalidad de "Base J2EE Support" conteste "OK" tras haber marcado la opción "Always enable capabilities and don't ask me again."



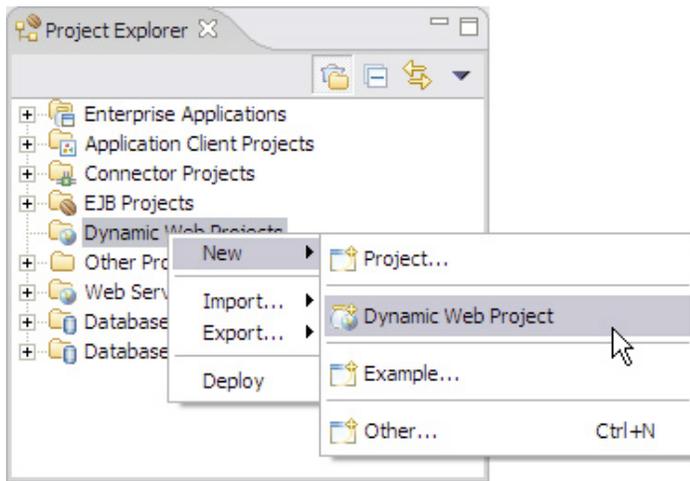
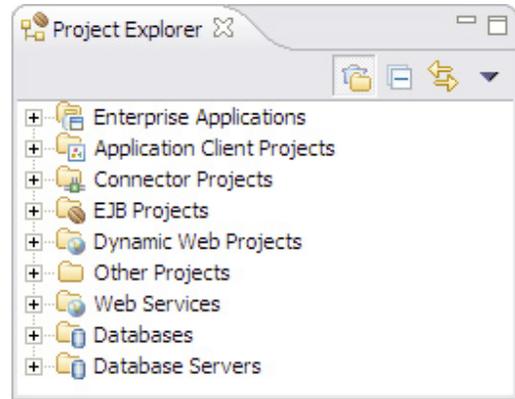
Tenga en cuenta que activar funcionalidad adicional aumenta los requerimien-

tos de memoria de WID.

Puede observar que hemos cambiado de perspectiva ya que del lado izquierdo, en lugar de la ventana de “Business Integration” ahora aparece la de “Project Explorer”.

Cambiar de una perspectiva a otra es tan sencillo como hacer click sobre el nombre de la perspectiva correspondiente junto al ícono 

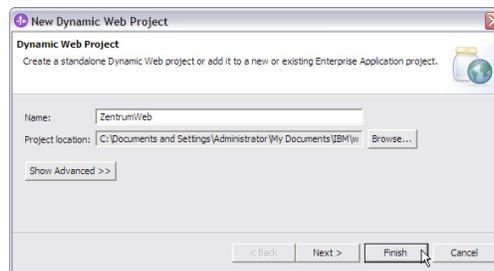
Ahora vamos a crear el proyecto de web dinámico que va a albergar nuestro servicio web. Para ello debemos hacer right-click sobre “Dynamic Web Projects” en la ventana de “Project Explorer” tal y como se muestra en la siguiente ilustración.



Elija un nombre para su nuevo proyecto de web dinámico. En nuestro caso lo hemos llamado “ZentrumWeb” para que no haya confusiones con el proyecto de integración que creamos anteriormente y al que llamamos “Zentrum”.

Pueden usar el diálogo para configurar toda una serie de atributos de su proyecto web, sin embargo en este caso no es necesario ya que no vamos a crear ninguna JSP, solo lo vamos a utilizar para crear un servicio Web. Por defecto el proyecto “Zentrum” se va a ejecutar en el servidor de aplicaciones por defecto, que es el mismo en el que se va a ejecutar el proceso y que es de tipo “WebSphere Process Server”. Normalmente los servicios deberían correr en otras máquinas separadas, sobre un servidor de aplicaciones más económico, ya sea usando

WebSphere Application Server ND (Network Deployment) o XD (eXtended Deployment).



Como hemos creado un nuevo proyecto de web, el entorno de desarrollo nos pregunta si queremos cambiar a la perspectiva de Web. No es algo necesario ya que no vamos a crear JSPs que requieren los editores de esta perspectiva, sin embargo para conocer las distintas perspectivas vamos a contestar que si, tal y como se muestra a continuación.

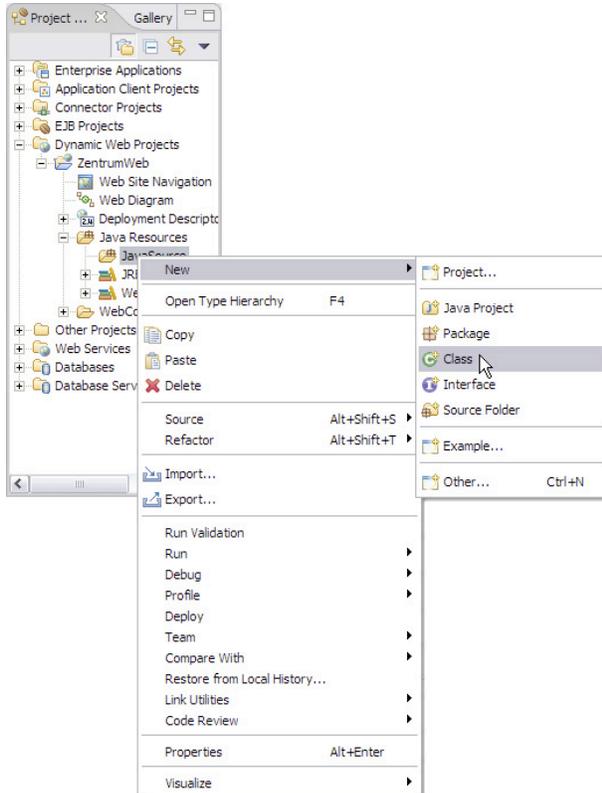


Seleccione la opción “Remember my decision” y pulse el botón “Yes”.

Ahora ya tenemos tres perspectivas habilitadas. Para poder ver el nombre de cada una de ellas debe extender la zona en la que se despliegan sus nombres, tal y como muestra la siguiente ilustración.

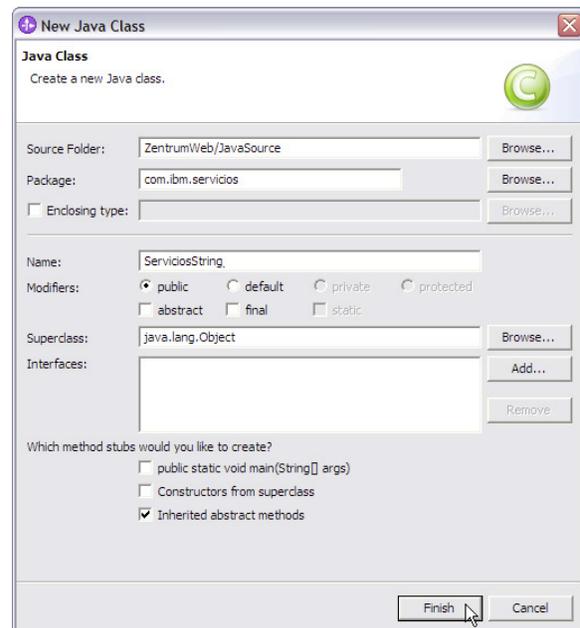


Ahora vamos a crear el Java Bean que va a contener el código de nuestro servicio web.



Haga right-click sobre “JavaSource” dentro de “Java Resources” y seleccione “New >> Class” dentro del menú contextual.

Llame la nueva clase “ServiciosString” y colóquela en el paquete com.ibm.servicios.

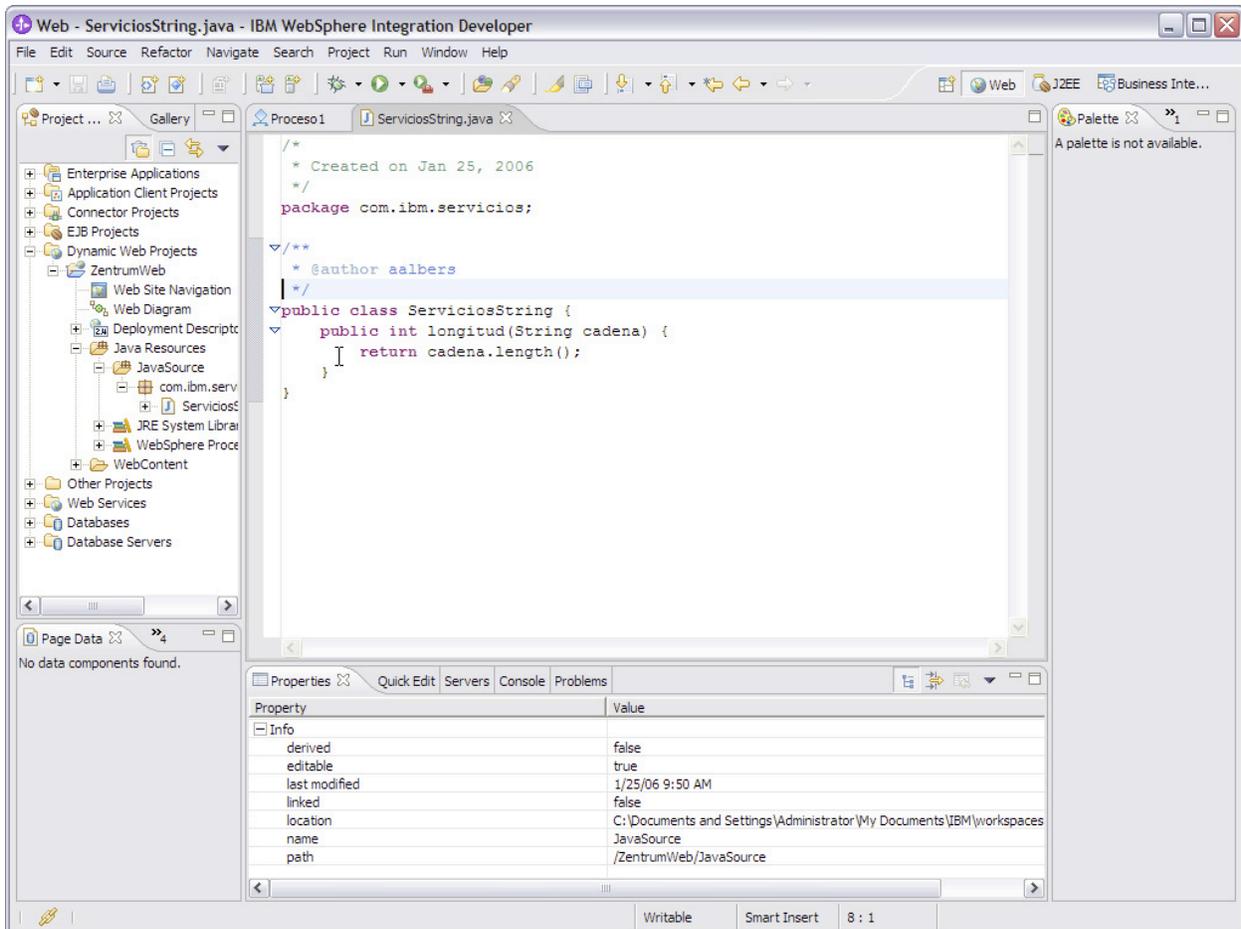


Pulse el botón “Finish” para terminar y crear la nueva clase.

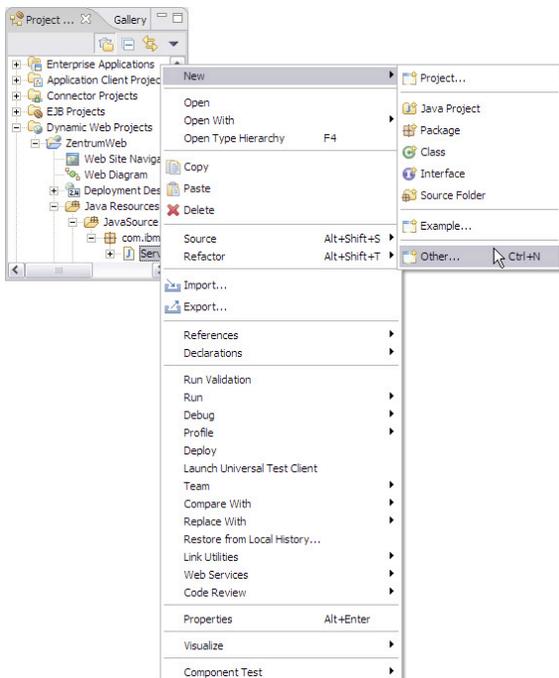
Modifique el código fuente de esta clase para que sea como el que se lista aquí:

```
package com.ibm.servicios;
public class ServiciosString
{
    public int longitud(String cadena)
    {
        return cadena.length();
    }
}
```

Si lo desea puede eliminar los comentarios que contienen la anotación TODO para que no aparezcan en la lista de tareas pendientes.

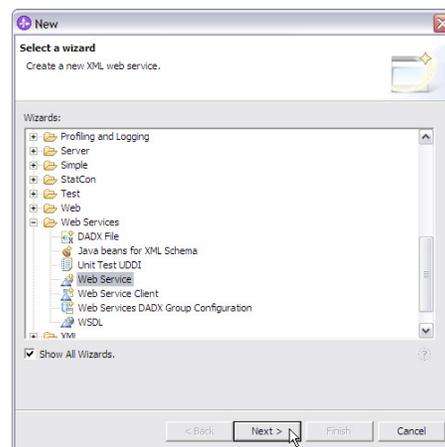


Salve el archivo usando la combinación de teclas Ctrl-S y cierre la pestaña.

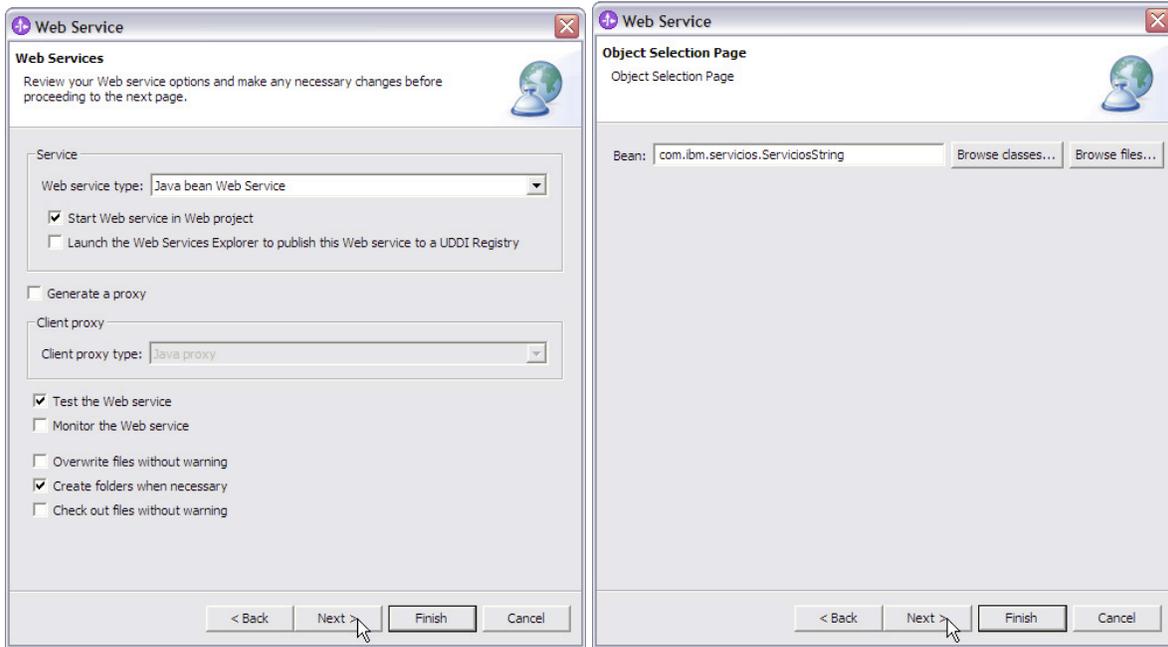


Convertir esa clase Java en un web service es una operación trivial como verá a continuación. Haga right-click sobre el nombre de la clase “ServiciosString.java” en la ventana de “Project Explorer” y seleccione “New >> Other...”.

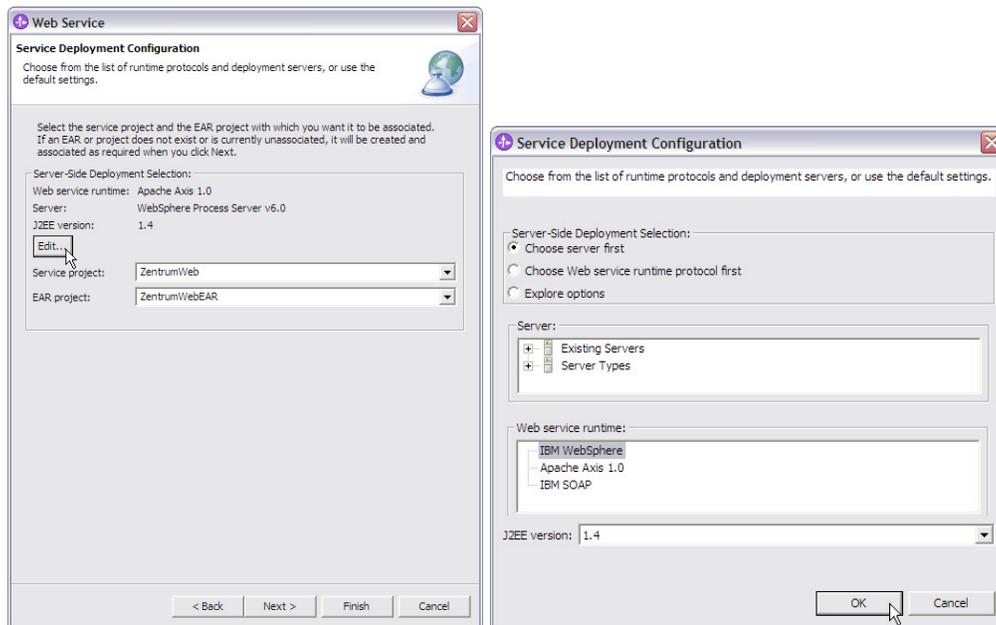
Marque la opción “Show All Wizards” y seleccione “Web Service” de la carpeta “Web Services”.



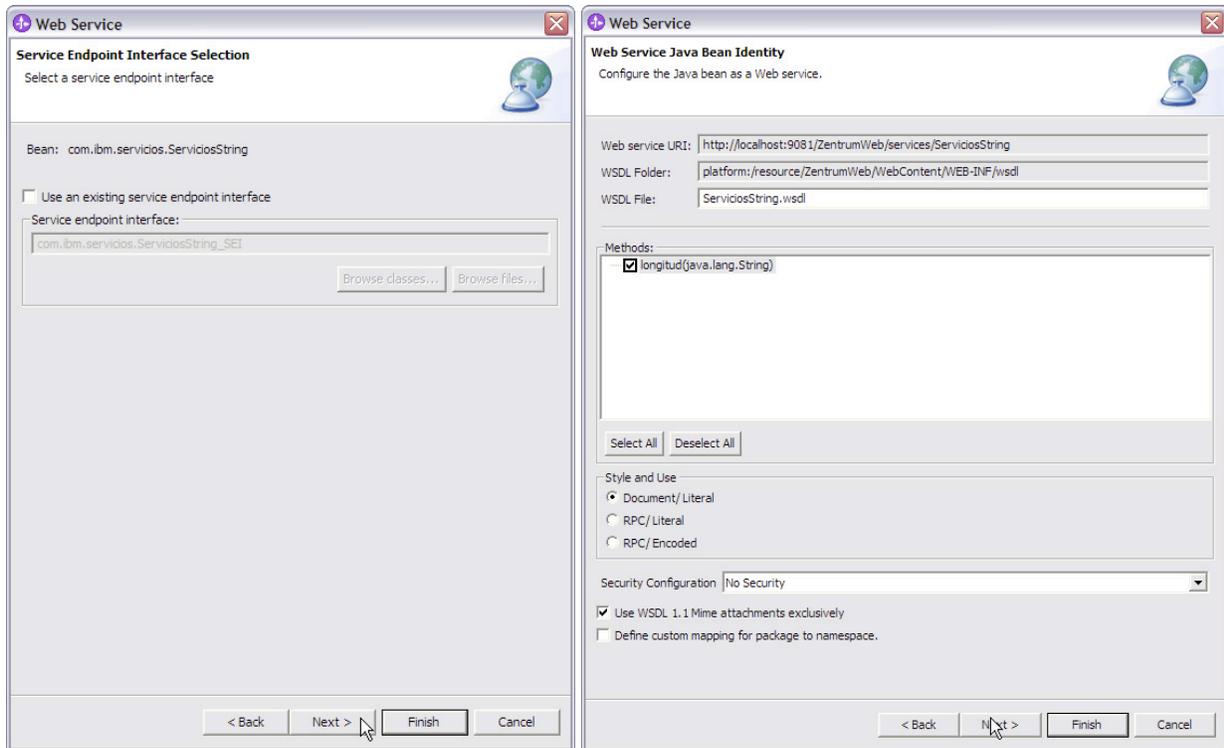
Pulse “Next”. Las opciones por defecto están correctas, no necesita hacer cambios. En la siguiente pantalla marque la opción “Test the Web service” y pulse “Next” tal y como se muestra en la siguiente ilustración. La siguiente pantalla no requiere realizar cambios, simplemente pulse “Next”.



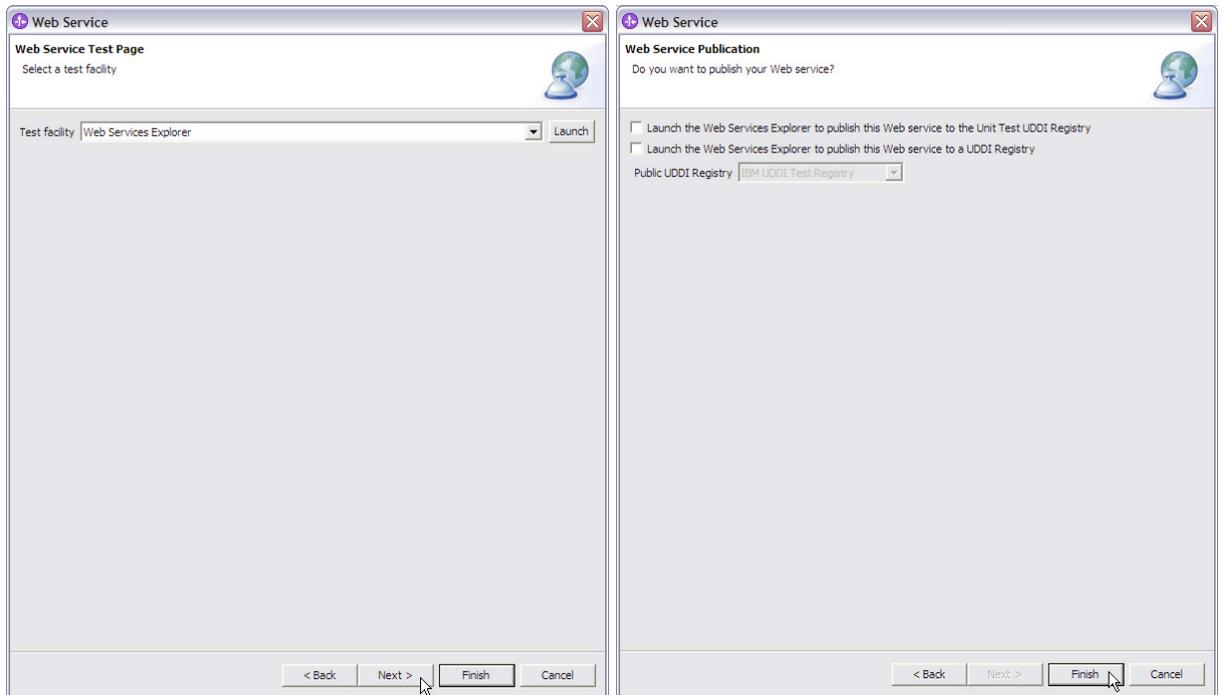
Para este servicio web vamos a usar el runtime de WebSphere en lugar de Apache Axis. Para ello es necesario pulsar el botón de “Edit...” y seleccionar el runtime adecuado en la pantalla que aparece y que se muestra a continuación. Pulse “OK” para aceptar el cambio y luego “Next” para pasar a la siguiente pantalla del wizard de creación de Servicios Web.



La siguiente pantalla no requiere de cambios así que simplemente pulse “Next”. La siguiente tampoco, pero observe cómo el método “longitud”, el único de la clase “ServiciosString” está seleccionado y se va a convertir en un servicio web, antes de pulsar el botón “Next” para continuar.

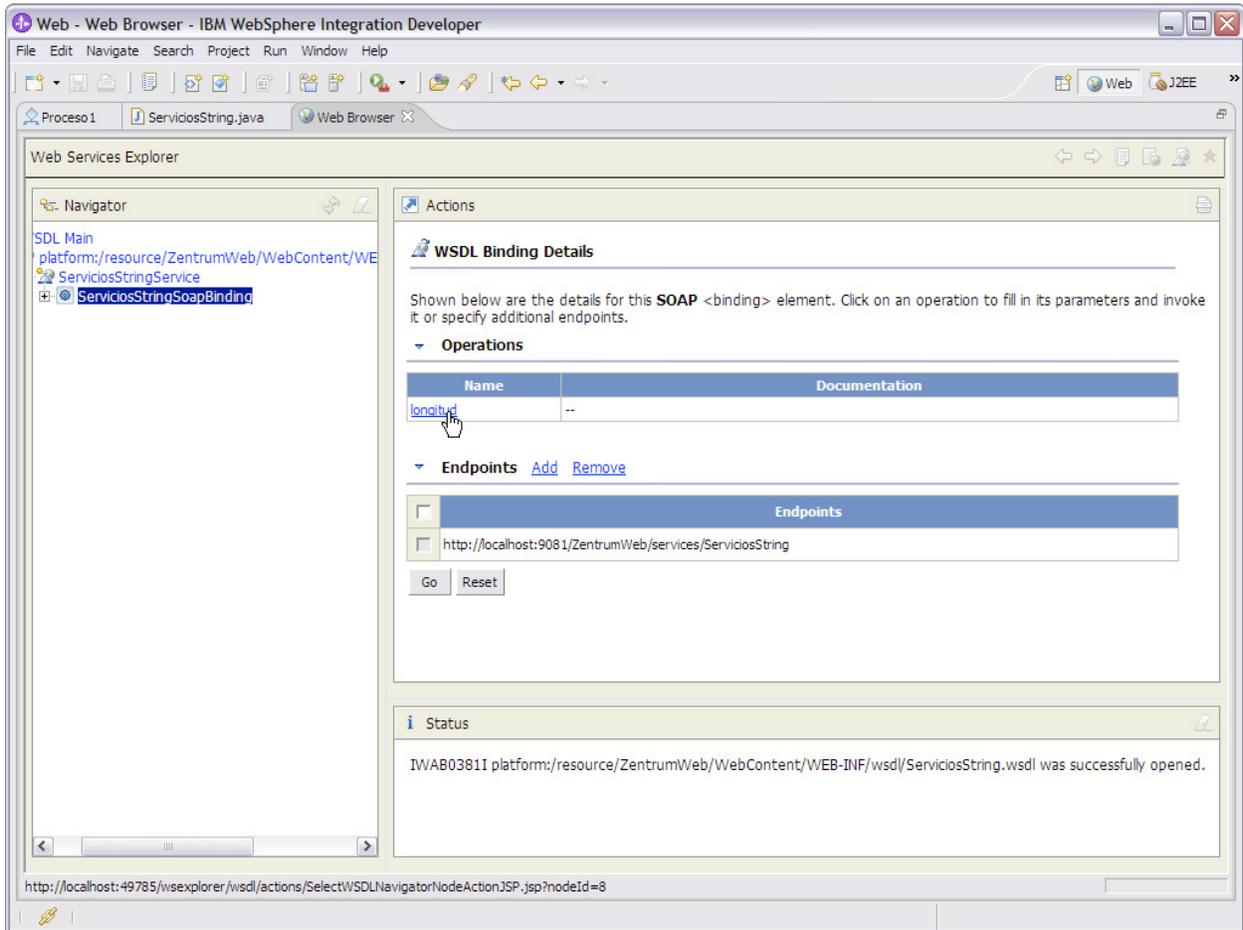


Las siguiente pantalla tampoco requiere realizar cambios. Pulse “Next” para pasar a la última pantalla del “Wizard”.

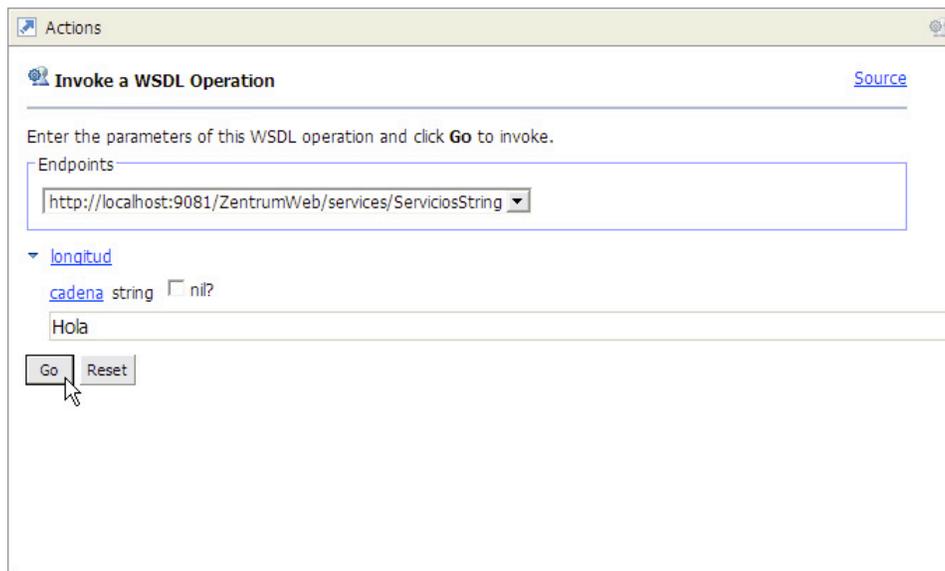


Pulse el botón “Finish” para que se genere el servicio web a partir del Java Bean. El proceso tarda un poco porque además de generar los archivos necesarios, el procedimiento arranca el servidor de aplicaciones tal y como se puede observar en la consola.

Tras crear el servicio, WID abre automáticamente el “Web Services Explorer” para probar el servicio recién creado. Para maximizar la ventana, haga doble click sobre la pestaña “Web Browser”.



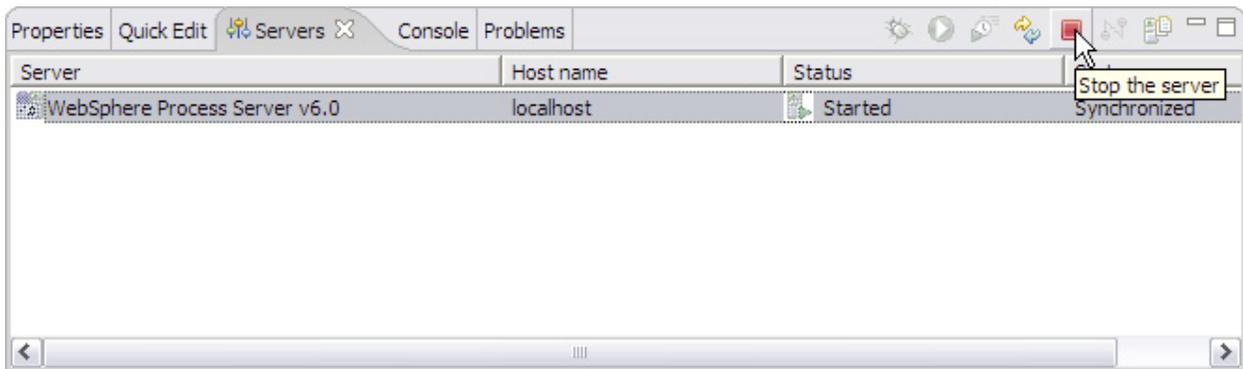
Haga click sobre “longitud” para probar el servicio. Entre una cadena cualquiera, por ejemplo “Hola” y pulse “Go”.



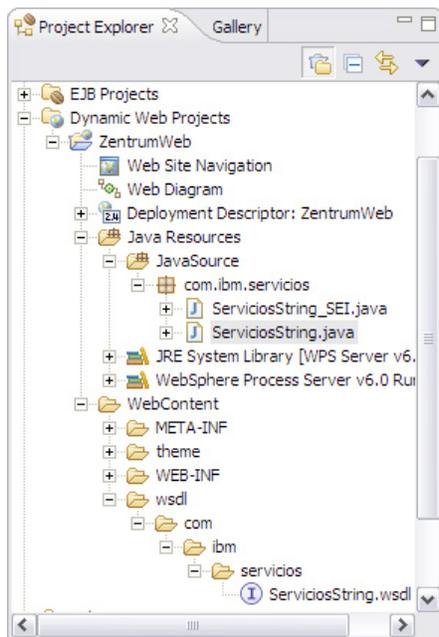
Si todo funciona correctamente debe ver en la ventana inferior el resultado de la operación, por ejemplo “4”.



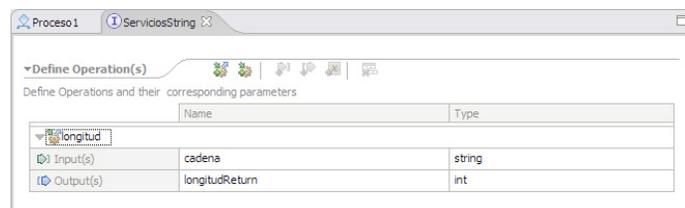
Una vez probado el servicio web, para ahorrar memoria vamos a detener el servidor de aplicaciones. Para ello haga click sobre la pestaña “Servers”. Seleccione el servidor “WebSphere Process Server v6.0 y deténgalo haciendo click sobre el ícono rojo tal y como se muestra a continuación.



Para crear el servicio web a partir de un Java Bean, WID debe crear varios archivos. El más interesante para nosotros es el archivo ServiciosString.wsdl. Este archivo es un archivo WSDL (Web Services Definition Language). Este tipo de archivos se utiliza para definir un conjunto de servicios web. En nuestro caso, como el Java Bean solo contenía un método, el archivo WSDL correspondiente solo define un servicio web. Cada archivo WSDL se divide en cuatro partes (Service, Bindings, Port types, Messages). Normalmente trabajaremos con los dos últimos, que definen los métodos así como los datos de entrada y salida que requieren. En algunos casos necesitaremos también trabajar con los otros dos que indican dónde residen los servicios y los transportes que se pueden utilizar para invocarlos.

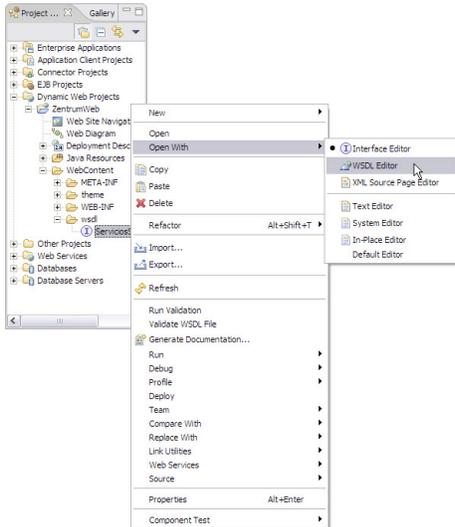


Por esta razón WID ofrece dos editores para examinar los archivos WSDL. Por defecto está seleccionado el el “Interface Editor”. Haga double-click sobre el archivo ServiciosString.wsdl para editar el archivo utilizando este editor. Debería aparecer una pantalla como la que se muestra a continuación.



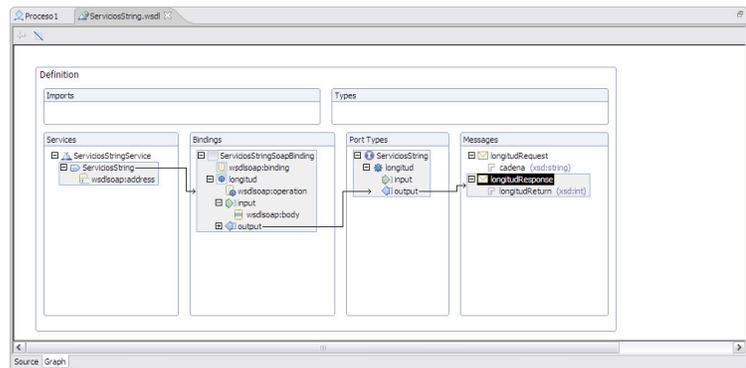
Observe como el archivo define una operación llamada longitud que recibe como parámetro de entrada un parámetro “cadena” de tipo “string” y devuelve un parámetro “longitudReturn” de tipo “int”. Los servicios web utilizan para la definición de datos XML-Schema, por lo tanto string se refiere a xsd:string e int a xsd:int.

Cierre el documento ServiciosString.wsdl haciendo click en la “x” que aparece junto al nombre del archivo en su pestaña.



Elegir el editor “WSDL editor” requiere hacer right-click sobre el archivo ServiciosString.wsdl y elegir “Open With >> WSDL editor”.

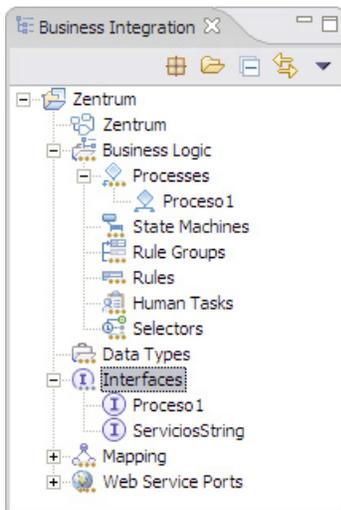
Como pueden observar, este editor es más completo que el anterior. Sin embargo, cuando trabajemos en la definición del proceso BPEL solo nos preocupa el interface del servicio, no dónde se ejecuta o el tipo de transporte que usaremos para comunicarnos con el servicio. Eso es algo que tendremos que definir más tarde. De esa manera, si cambia el tipo de transporte o la ubicación del servicio, eso no afecta al proceso.



Cierre el documento ServiciosString.wsdl haciendo click en la “x” que aparece junto al nombre del archivo en su pestaña.

Al abrir el documento WSDL con el editor WSDL hemos cambiado el editor que WID usa por defecto. Como en realidad vamos a trabajar casi siempre con el “Interface Editor”, tenemos que indicar que ese es el que queremos usar por defecto.

Haga right-click sobre el documento ServiciosString.wsdl y elija “Open With >> Interface Editor”. Cierre el documento ServiciosString.wsdl haciendo click en la “x” que aparece junto al nombre del archivo en su pestaña.

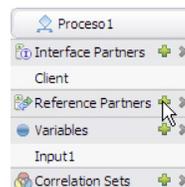


Ahora vamos a copiar el archivo ServiciosString.wsdl a nuestro proyecto de integración para poder usarlo en nuestro proceso. Haga right-click sobre el archivo y seleccione “Copy”.

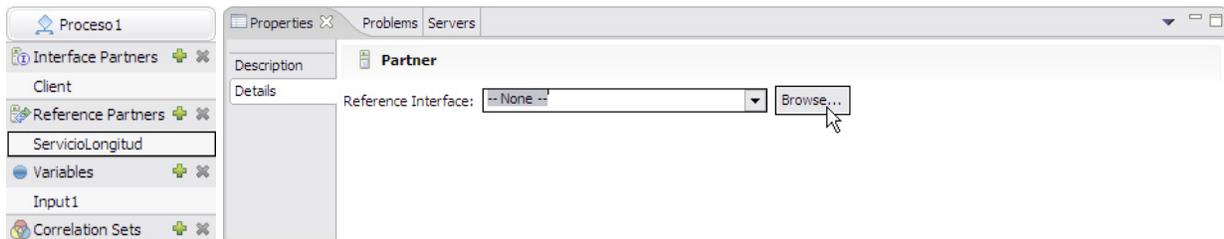
Cambie a la perspectiva de “Business Integration”.

Haga right-click sobre “Interfaces” y pulse “Paste”.

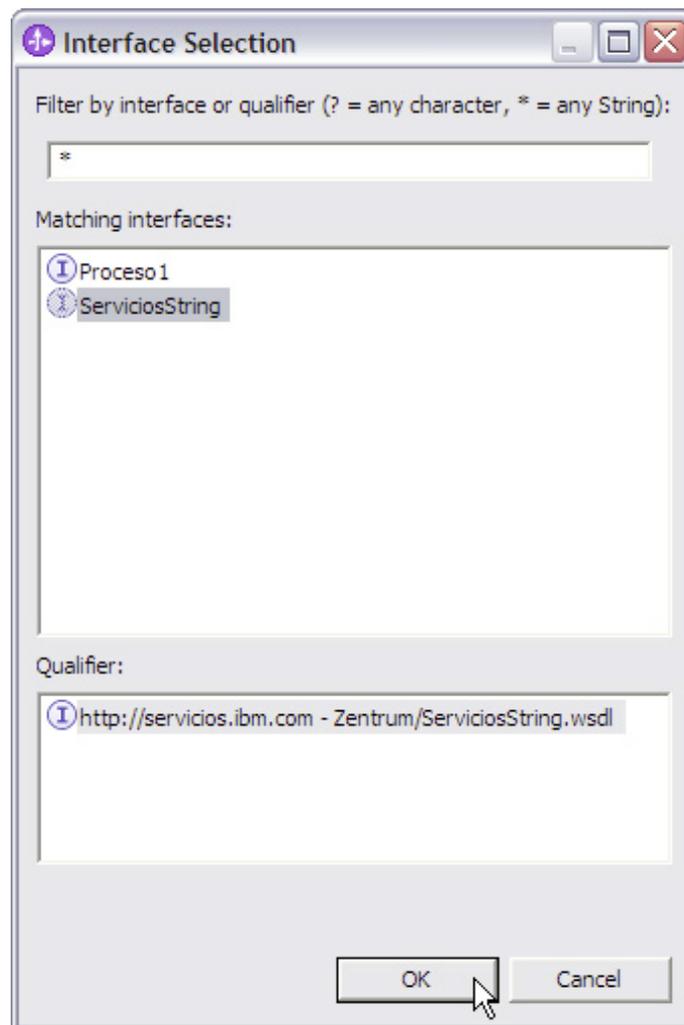
Ahora vamos a hacer que este interface pueda ser utilizado en el proceso. Haga click sobre el símbolo “+” junto a “Reference Partners”.



Escriba “ServicioLongitud” en lugar de “Partner”. Tenga cuidado de no usar espacios en el nombre. Con el elemento recién creado seleccionado, observe la ventana inferior.

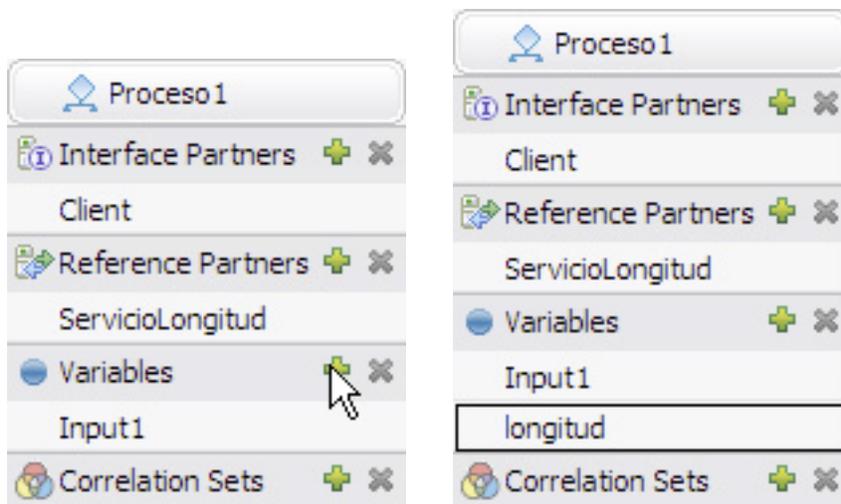


Pulse el botón de “Browse” para seleccionar una interface WSDL. Esto es lo que permitirá a WID saber qué operaciones ofrece el “Partner”, en nuestro caso la operación longitud.



Seleccione la interfaz “ServiciosString” y pulse el botón “OK” para continuar.

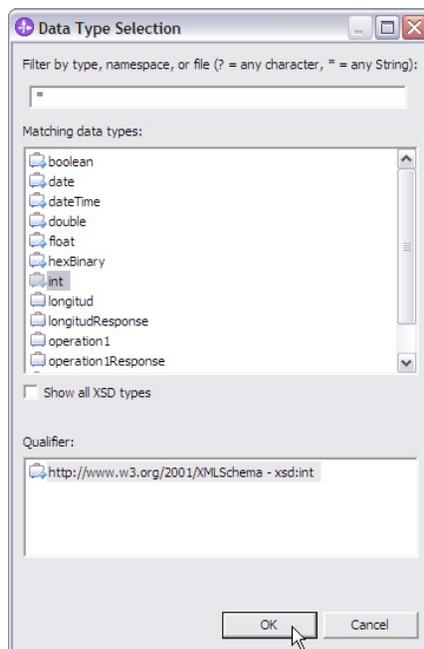
El servicio “longitud” regresa un resultado de tipo `xsd:int` y por lo tanto necesitamos crear una variable de ese mismo tipo para poder almacenar el resultado de manera temporal. Llamaremos a esa variable “longitud”. Para crearla haga click sobre el símbolo “+” junto a “Variables” e introduzca el nombre de la variable.



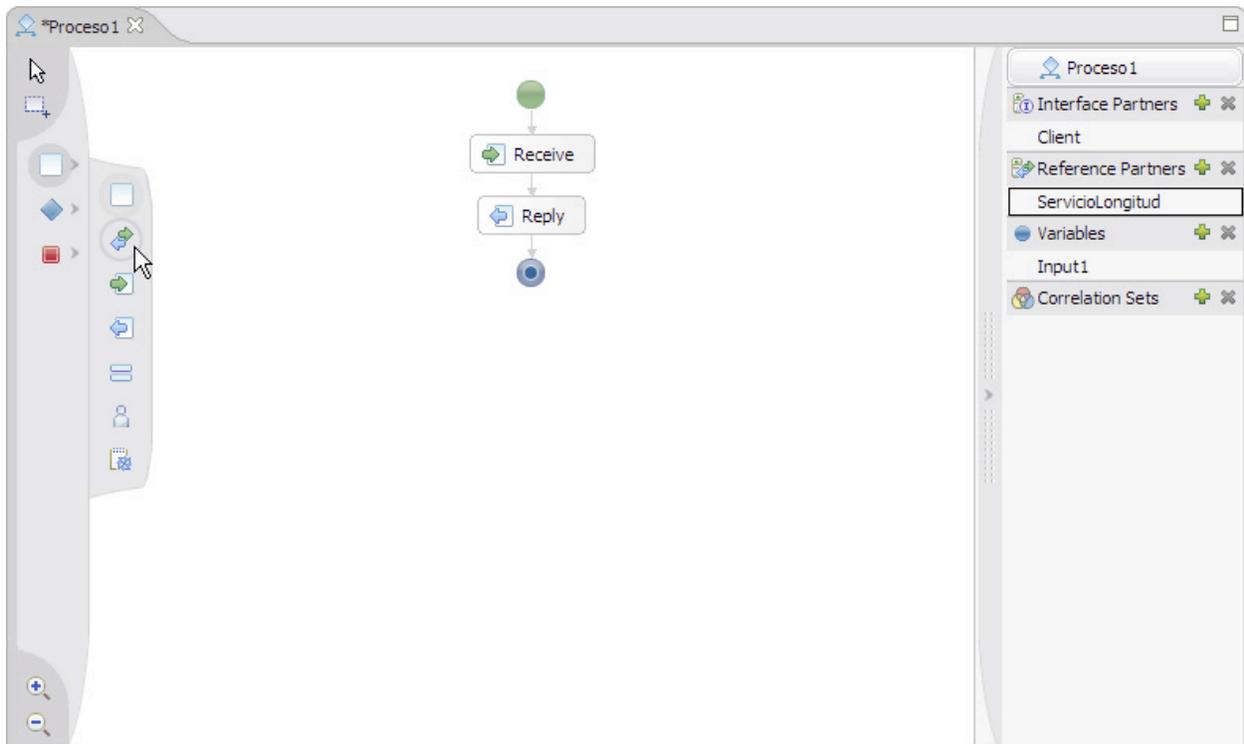
Con la variable “longitud” seleccionada observe la ventana inferior, la cual se muestra a continuación.



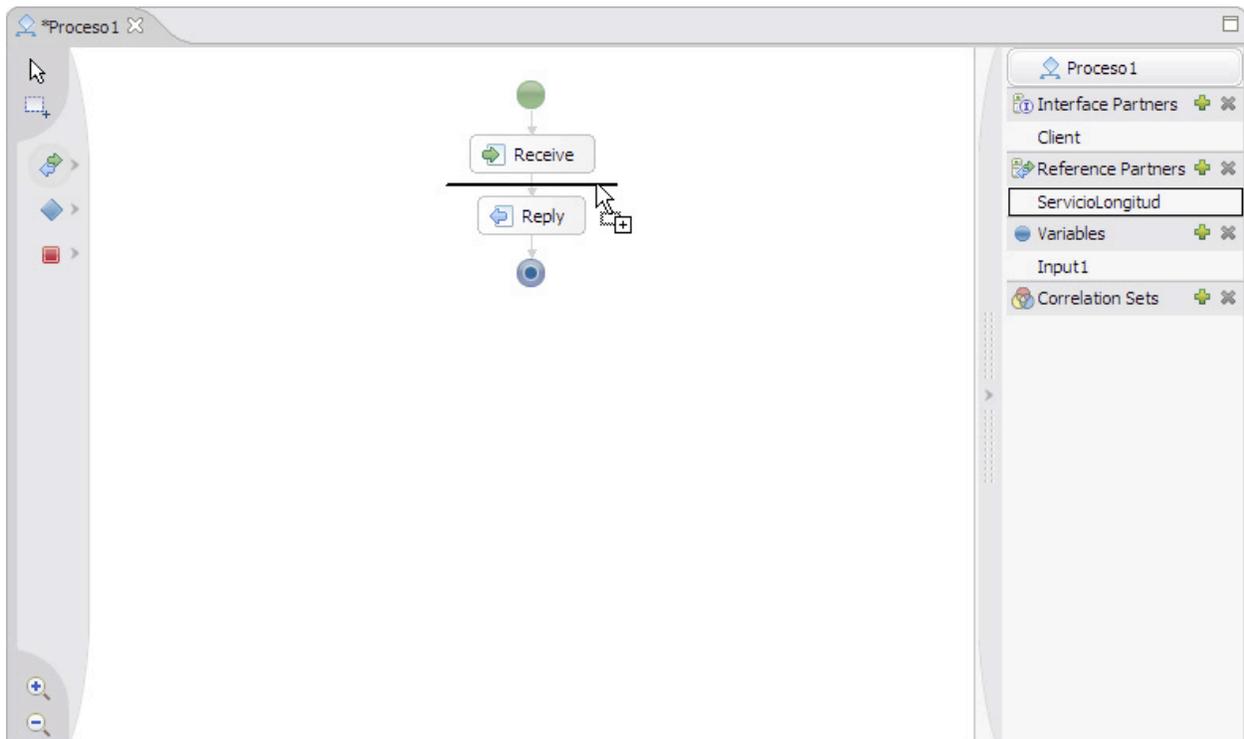
La variable “longitud” no tiene ningún tipo definido. Pulse el botón “Browse” para asignarle un tipo. Elija “int” y pulse el botón “OK”.



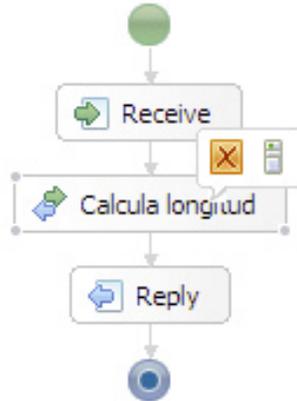
Ahora ya estamos listos para invocar el servicio "longitud" desde nuestro proceso. Para ello vamos a insertar una tarea de tipo "invoke" entre las actividades "Receive" y "Reply" que inician y terminan el proceso respectivamente. Para ello seleccione la tarea de la paleta, tal y como se muestra en la siguiente ilustración.



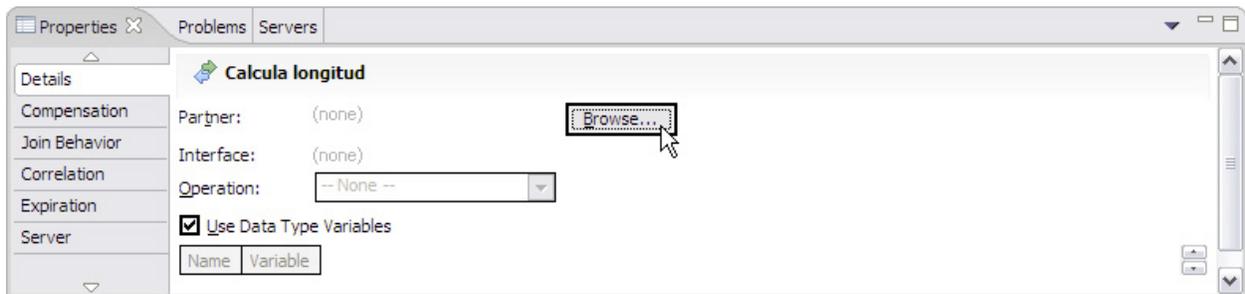
Mueva el cursor hasta posicionarse entre las tareas de "Receive" y de "Reply" como se muestra a continuación.



Haga click para insertar la tarea en el lugar deseado y cambie su nombre a “Calcula longitud”.



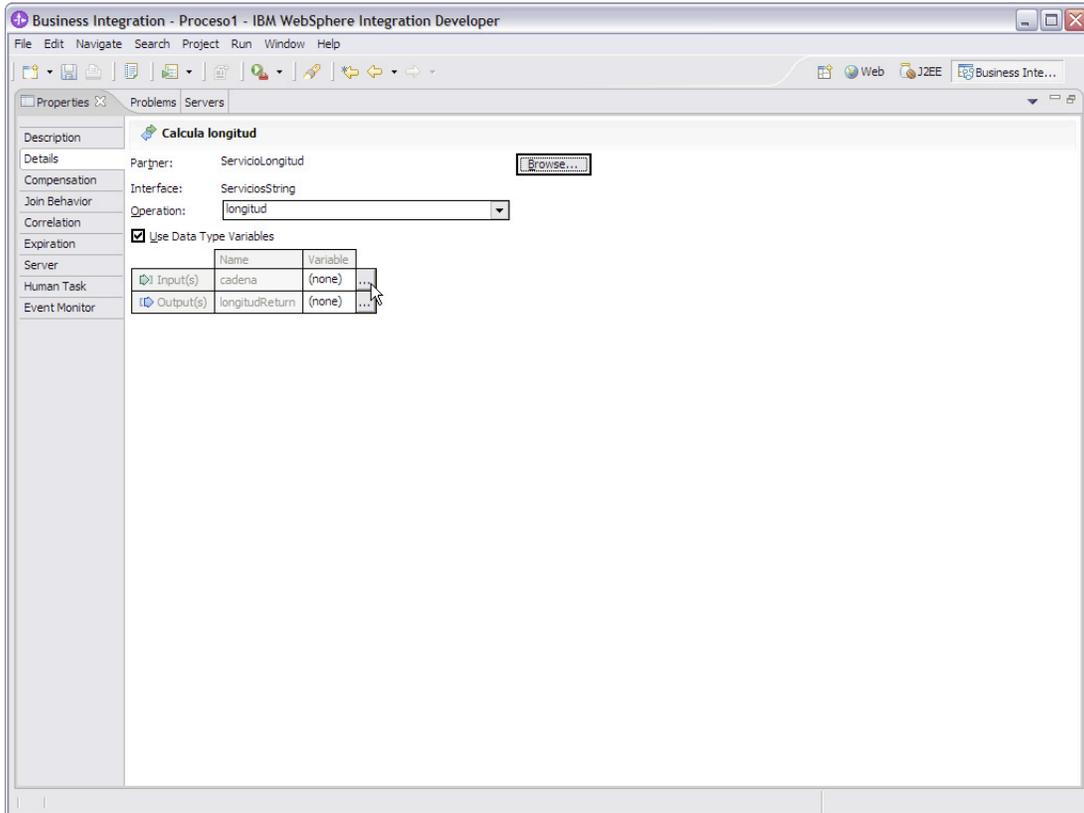
Ahora, con la tarea “Calcula longitud” seleccionada, observe la ventana inferior, en la pestaña de detalles. Vemos que la tarea no está ligada a ningún “partner” y por lo tanto tampoco tiene una operación definida.



Pulse el botón “Browse” para relacionar la tarea con el partner link creado anteriormente (ServicioLongitud) y pulse “OK” para continuar.

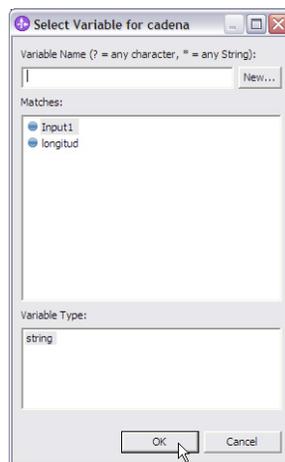


Haga doble-click sobre la pestaña “Properties” para que ocupe toda la pantalla y facilitar así las tareas de edición que tendremos que realizar a continuación.

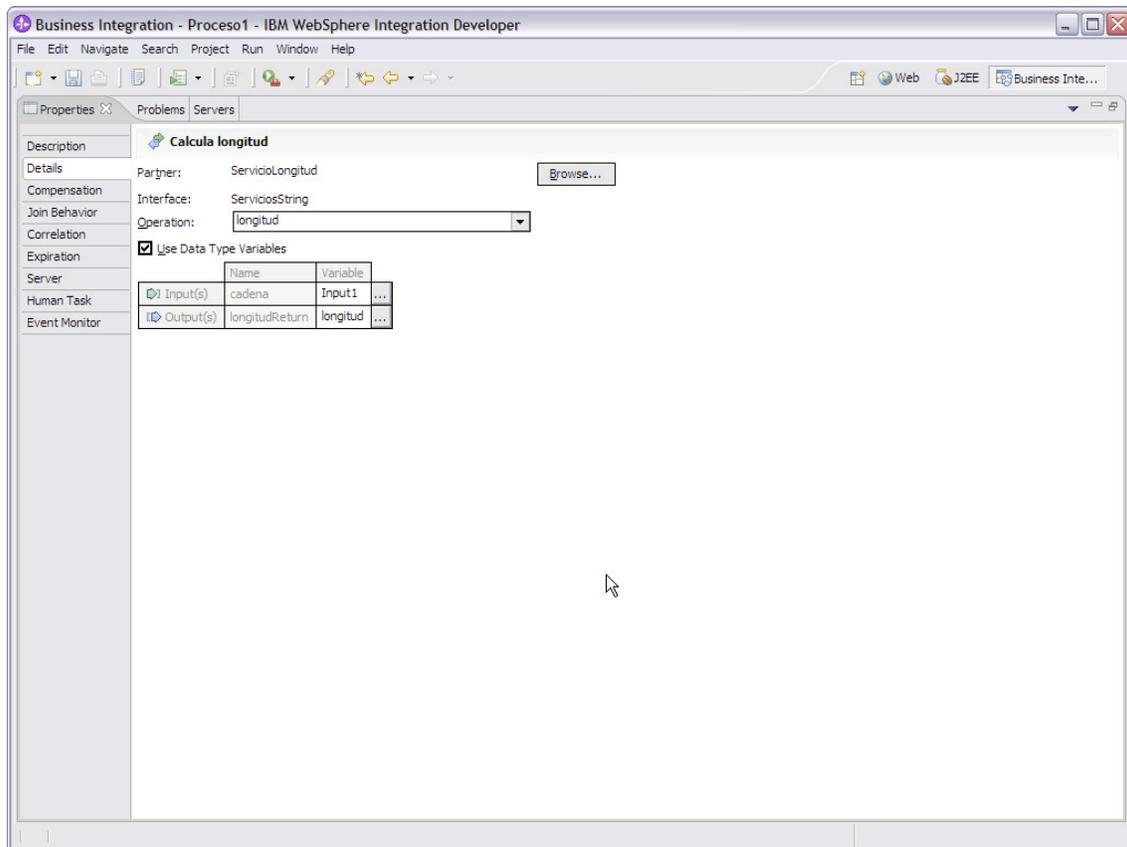


Vemos que la operación “longitud” ha sido seleccionada automáticamente, ya que es la única definida en Servicio-Longitud. Sin embargo, no hay variables asignadas para pasar la cadena de entrada y recibir el resultado de la operación. Tenemos que indicar a WID que la cadena de entrada está contenida en la variable “Input1” que recibe el proceso al inicio y que queremos que el resultado de la operación se almacene en la variable “longitud”.

Pulse el botón con puntos suspensivos (“...”) en el renglón correspondiente a “Input(s)”. Seleccione la variable “Input1” de la lista y pulse “OK”.

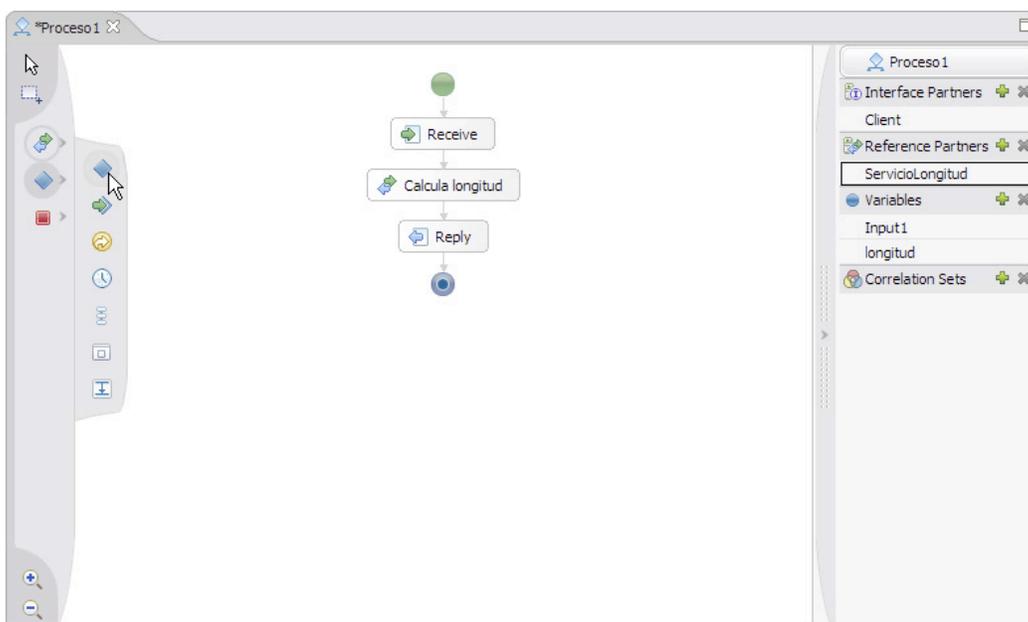


Repita la operación para asignar la variable “longitud” de tipo int en el renglón correspondiente a “Output(s)”.

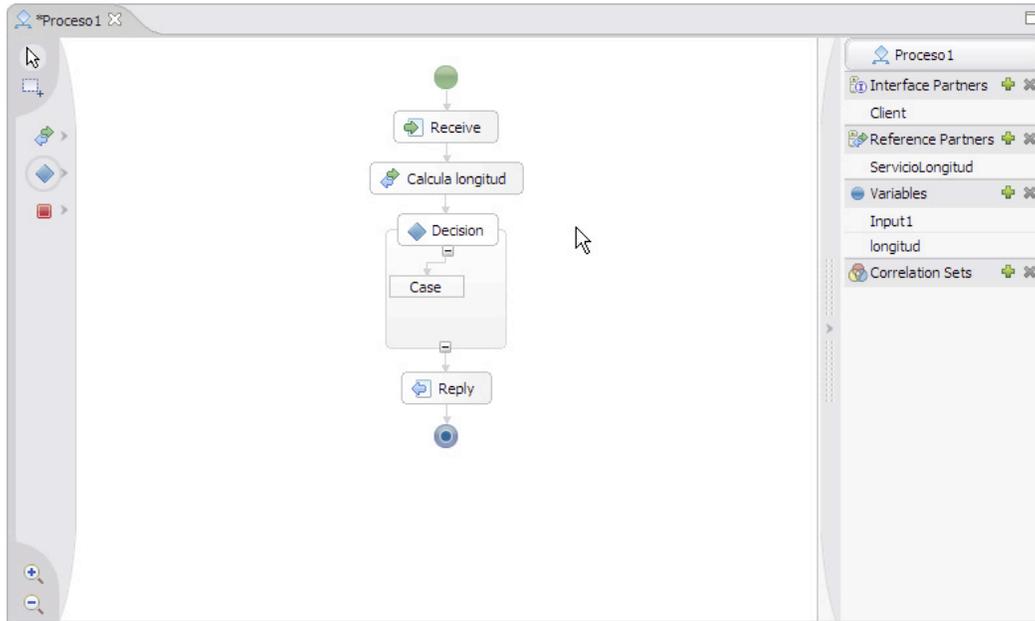


Conserve todos los cambios realizados pulsando Ctrl-S. Minimice la pestaña "Properties haciendo doble-click sobre el título de la misma.

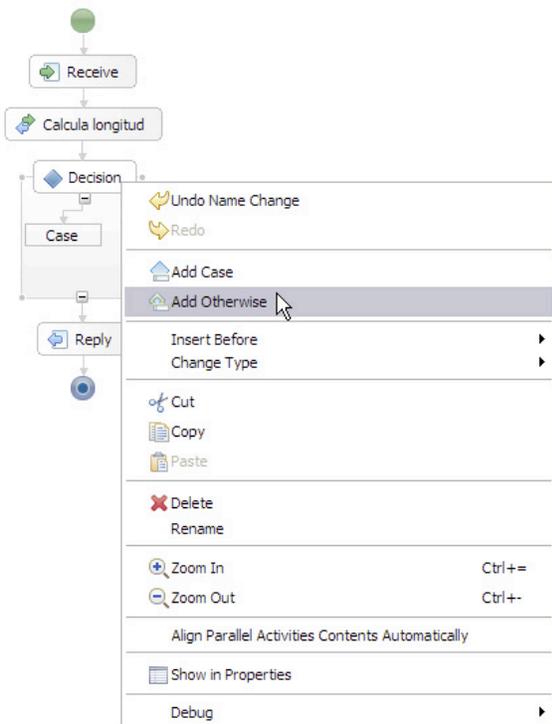
Ahora vamos a añadir una decisión a nuestro proceso. Dependiendo de si la longitud de nuestra cadena de caracteres es mayor o igual a ocho, regresaremos la cadena "La cadena de entrada es larga" o "La cadena de entrada es larga".



Seleccione de la paleta el elemento “Choice” el cual se utiliza para modelar una decisión. Posiciónelo tras la tarea “Calcula longitud”.

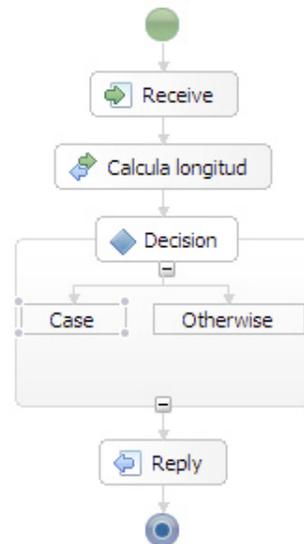


Por defecto una decisión es creada con un único elemento “Case”. Sin embargo, es posible añadir tantas ramas “Case” como sea necesario. Opcionalmente es posible añadir una rama “Otherwise” que solo se ejecutará si todas las ramas “Case” han dado un resultado negativo.



En este ejemplo vamos a verificar si la longitud de la cadena es inferior a ocho caracteres y de lo contrario regresaremos la cadena “La cadena de entrada es larga”. Por lo tanto, lo más sencillo es usar “Otherwise” para modelar ese caso.

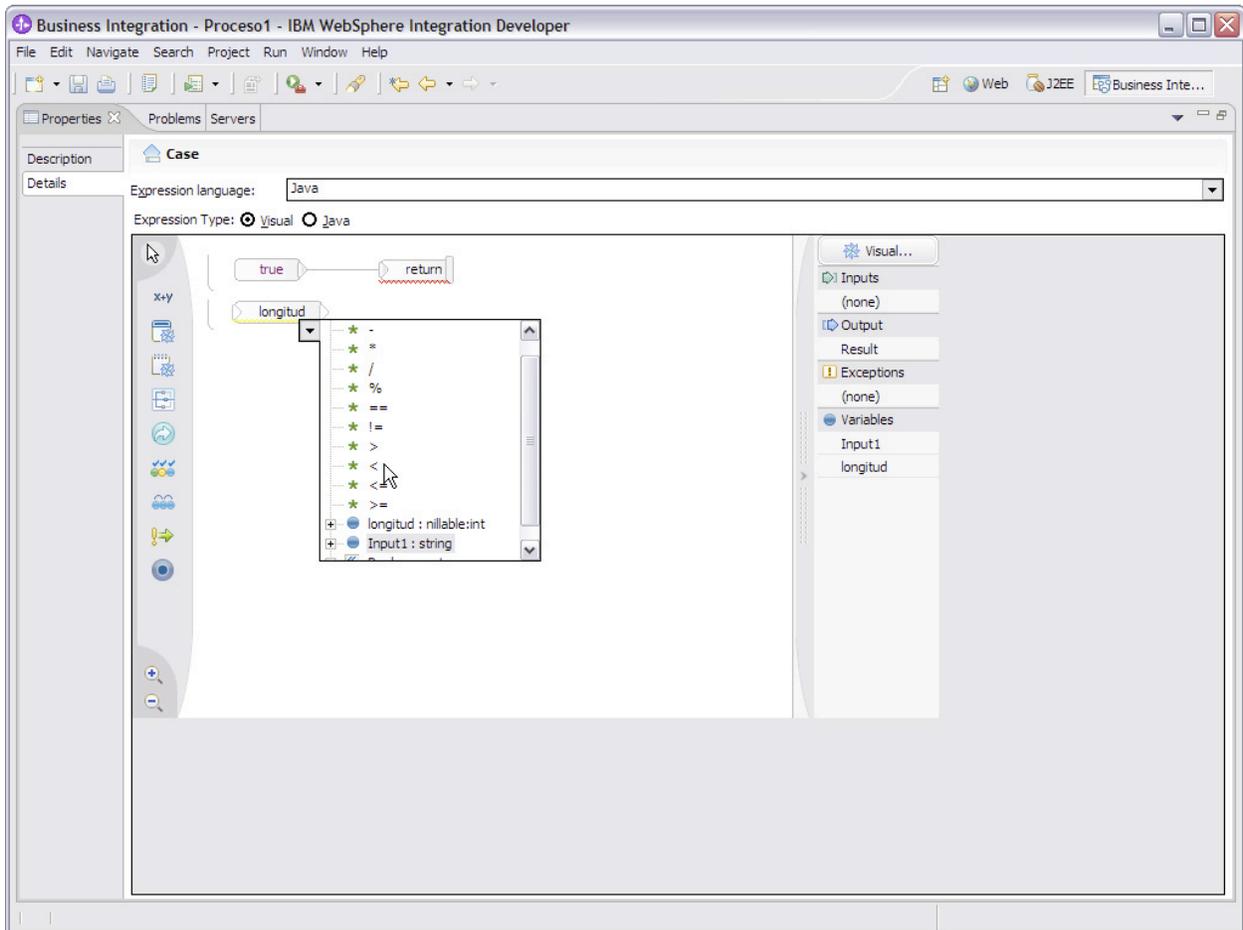
Para agregar una rama “Otherwise” a la decisión haga right-click sobre el ícono de decisión y elija la opción “Add Otherwise” del menú contextual.



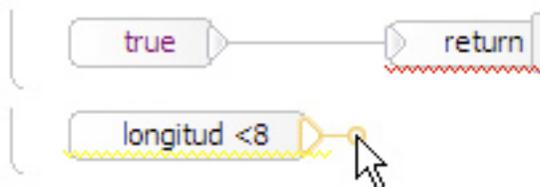
Ahora debemos modelar la expresión para la rama “Case” (longitud < 8). Haga click sobre la cajita “Case” para seleccionar el elemento. WID permite crear expresiones en distintos lenguajes. Si estuviéramos utilizando variables complejas que contienen XML, quizás lo aconsejable sería utilizar XPath. Sin embargo, como en este caso se trata de un entero es más sencillo hacerlo en Java. Ahora bien, recordemos que WID, a diferencia de Rational Application Developer está pensado para ser usado por personas que no necesitan conocer nada de Java. Por esta razón, WID ofrece un editor de expresiones Java gráfico, muy sencillo de usar.

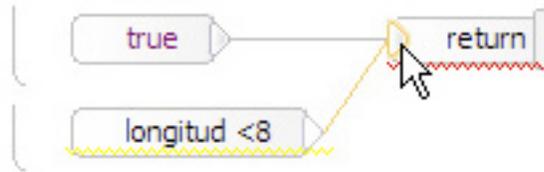
Antes de empezar, maximice la ventana inferior para poder editar cómodamente la expresión, haciendo doble-click sobre el título de la pestaña.

Arrastre la variable longitud debajo del valor “true”. Haga click justo detrás de la letra d al final de “longitud”. seleccione el operador de comparación “<”.



Seleccione la extremidad de la expresión recién creada y únala con el “return” arrastrando el cursor tal y como se muestra a continuación

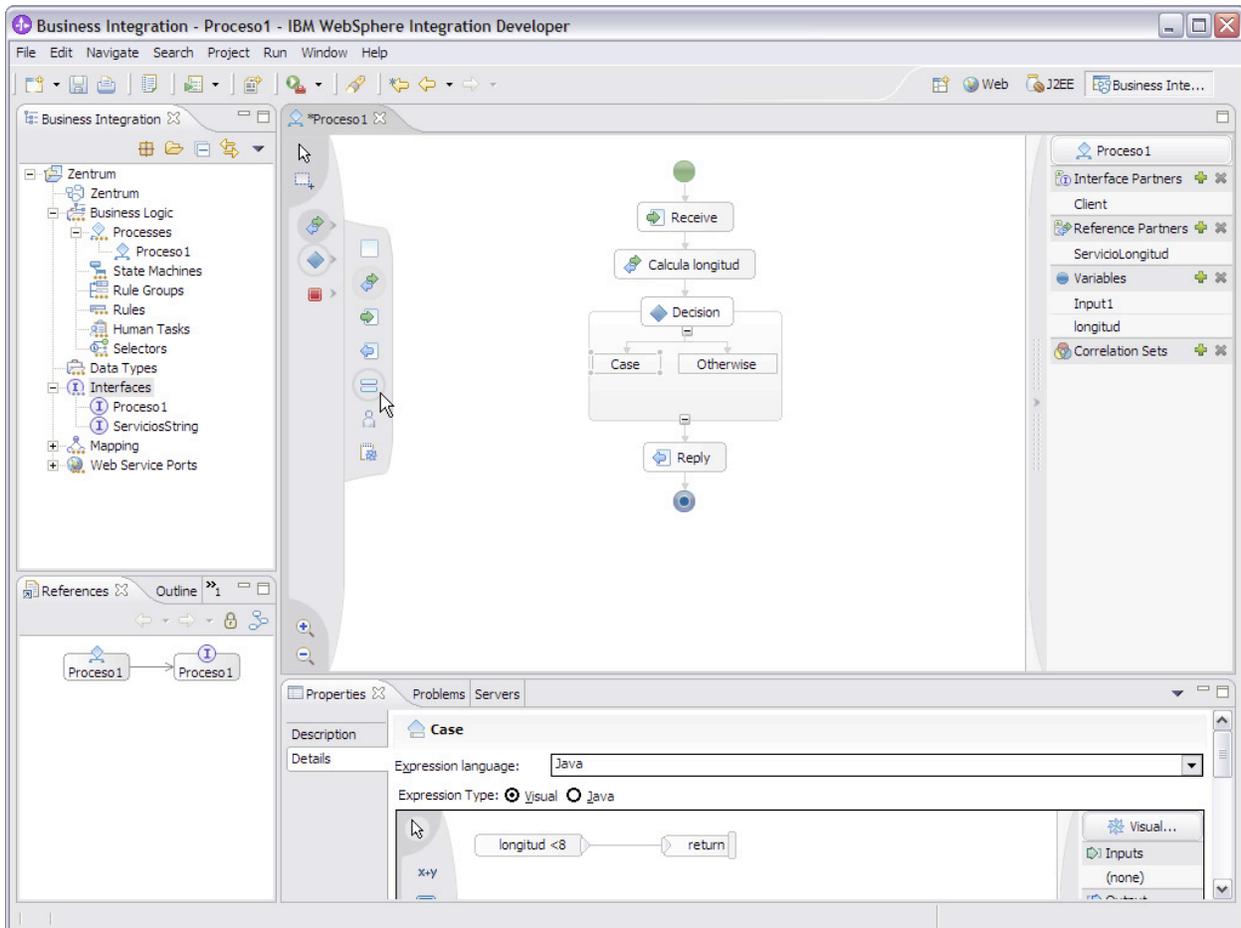




Seleccione la expresión “true” y pulse la tecla “Delete” para eliminarla.

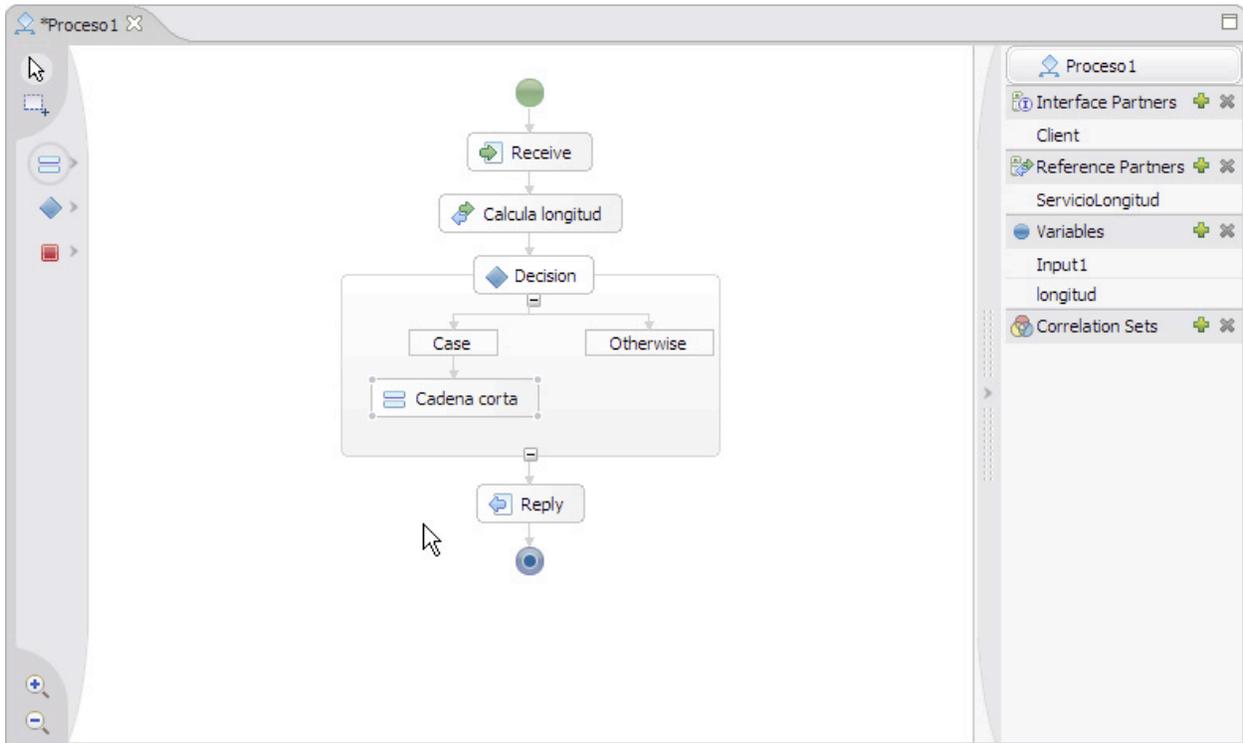


Una vez terminado, minimice la pestaña de “Properties” haciendo doble-click sobre su título. A continuación encontrará una ilustración de cómo debe estar su proyecto.

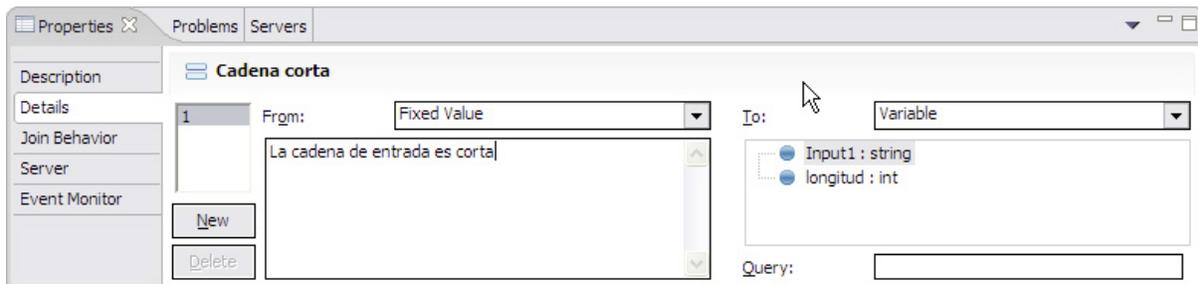


Ya solo nos queda un paso más para terminar de definir nuestro proceso BPEL. Dependiendo del resultado de la comparación debemos asignar valores distintos a la variable de salida (“Input1”). Para lograrlo usaremos la tarea “Assign” la cual se representa con un ícono en forma de signo de igualdad (“=”).

Seleccione la tarea “Assign” de la paleta y colóquela bajo la rama de “Case”. Cámbiele el nombre a “Cadena corta” tal y como se muestra a continuación.

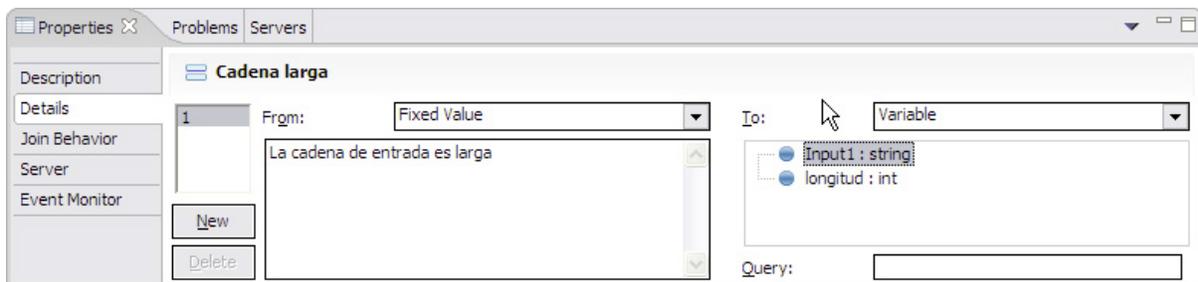


En la ventana inferior, defina la tarea de asignación tal y como se muestra a continuación. Vamos a asignar la cadena “La cadena es corta” (un valor fijo) a la variable “Input1”.



Ahora cree otra tarea de asignación bajo la rama “Otherwise” de la decisión “Decision”.

Al igual que para la primera tarea de asignación que definimos anteriormente, ahora debe definir la tarea de asignación en la ventana inferior. Asigne “La cadena es larga” a la variable “Input1”.



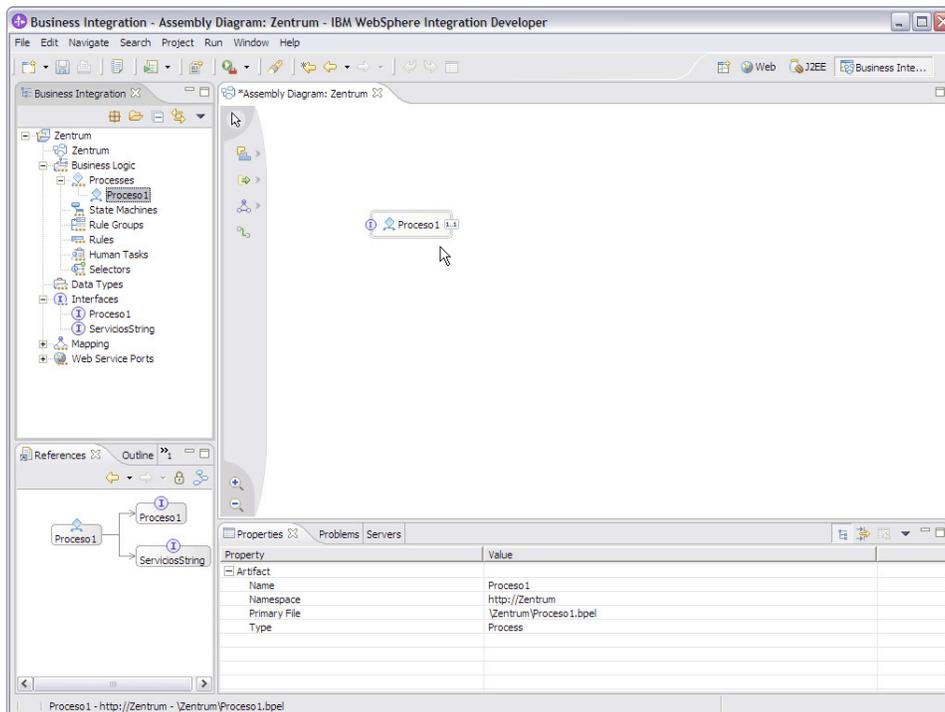


Con esto hemos terminado de crear nuestro flujo BPEL. Podemos salvarlo pulsando Ctrl-S. Cierre la ventana del “Proceso1” porque ya no la necesitamos.

Un proceso BPEL es un componente SCA (Software Component Architecture). Existen otros componentes SCA como Reglas, Tareas humanas, Selectores, etc. que también se crean con WID. Un proceso complejo puede consistir de decenas o incluso cientos de componentes SCA encadenados en una secuencia. En este laboratorio, nuestro proceso solo va a contener un componente SCA, el proceso BPEL “Proceso1” que acabamos de crear.

Haga doble-click sobre “Zentrum”. Se trata de lo que se conoce como “Assembly diagram” en el que vamos a crear nuestro proceso.

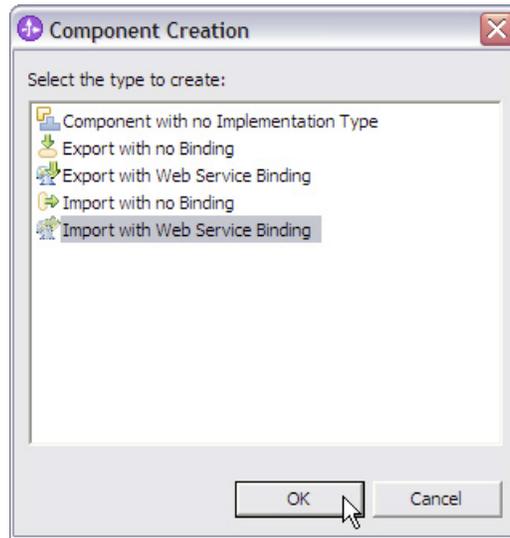
Arrastre “Proceso1” desde la ventana de “Business Integration” sobre la ventana “Zentrum” que acaban de abrir.



El proceso "Proceso1" invoca un servicio externo, por lo que debemos dar al componente la información necesaria de dónde se encuentra ese servicio y cómo invocarlo, en forma de un archivo WSDL. Si el proceso invocara más servicios externos, tendríamos que proporcionar un archivo WSDL por cada uno de ellos.

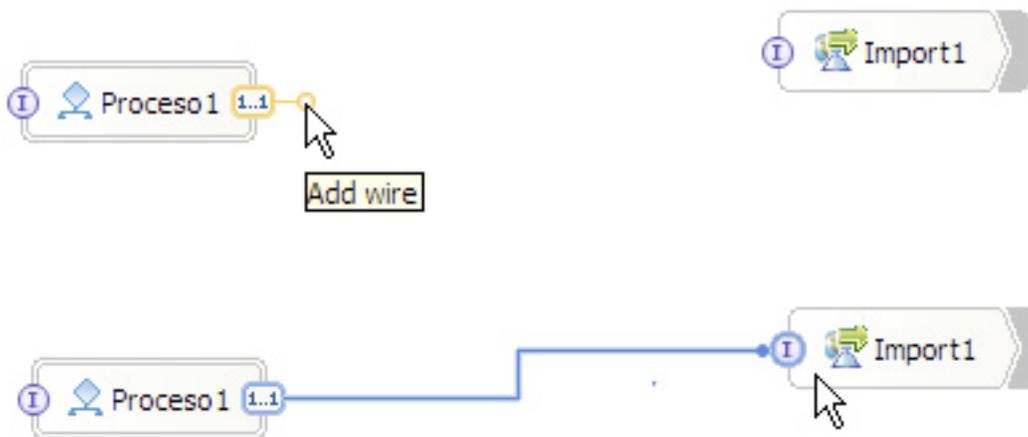
Ahora arrastre el interfaz "ServiciosString" desde la ventana de "Business Integration" sobre la ventana "Zentrum", a la derecha de dónde colocó el "Proceso1".

WID abre una ventana para saber cómo importar el interfaz.



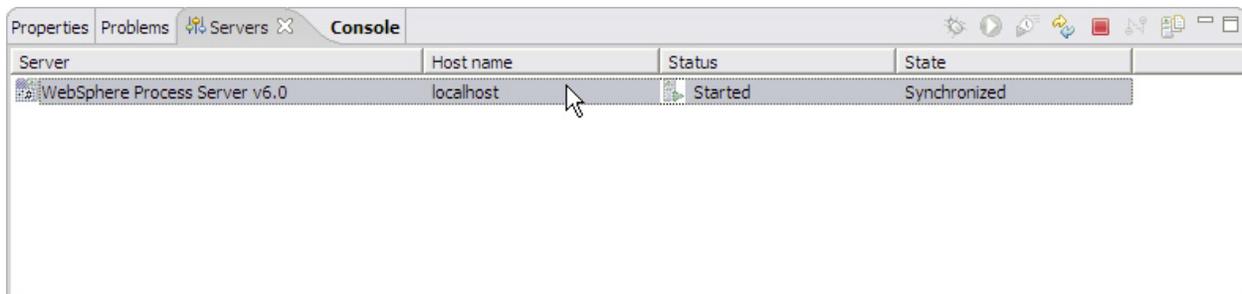
En este caso, como se trata de un servicio web al que se va a conectar nuestro proceso, elegimos "Import with Web Service binding".

Ahora debemos conectar el proceso "Proceso1" al componente que acabamos de crear.

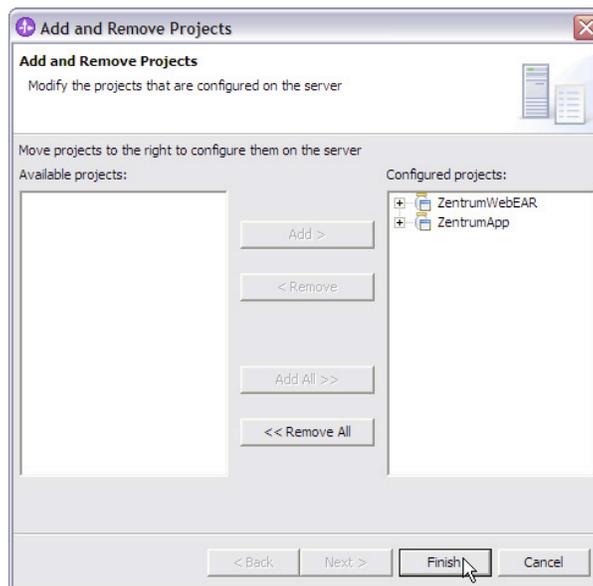
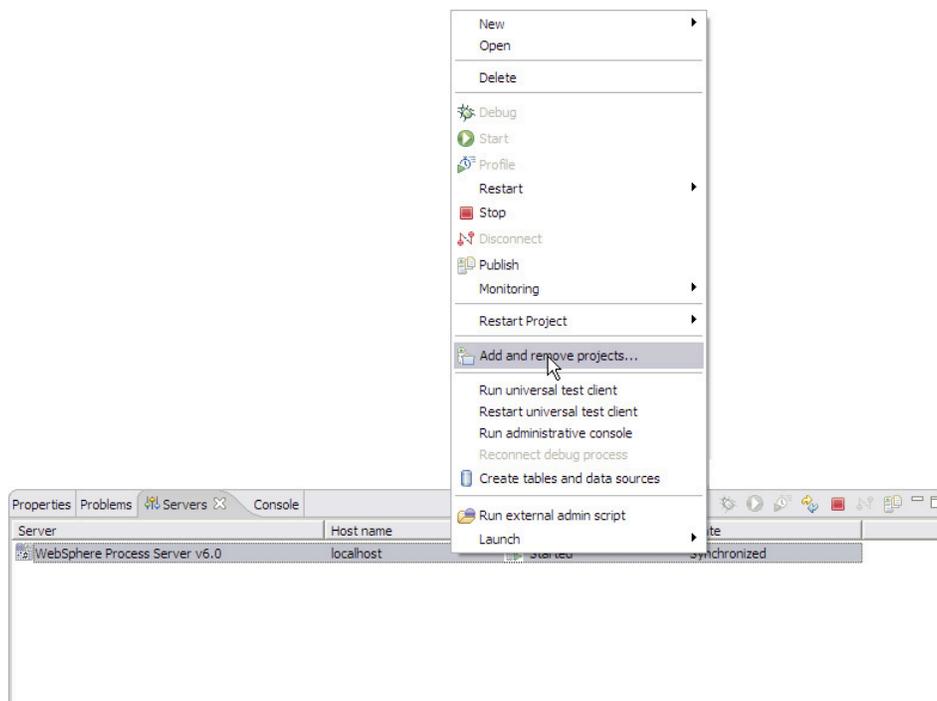


Ya estamos listos para poder probar el proceso. Salve los cambios utilizando la combinación de teclas Ctrl-S.

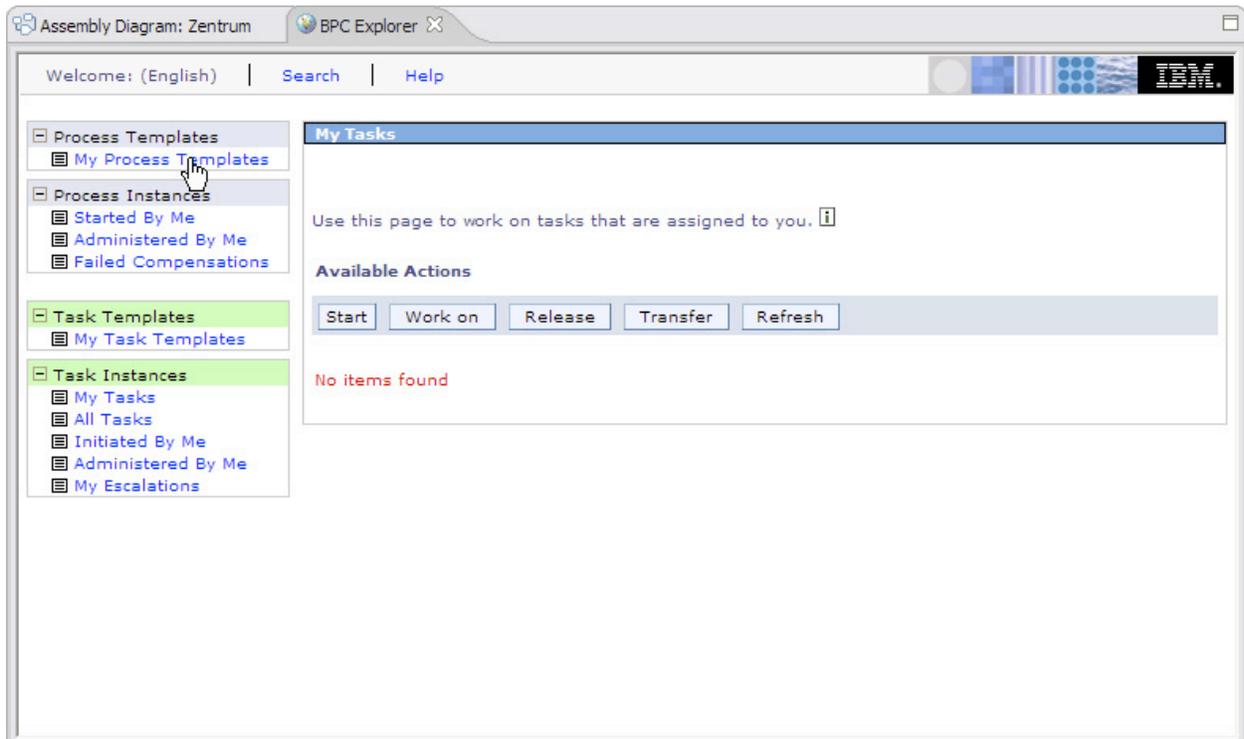
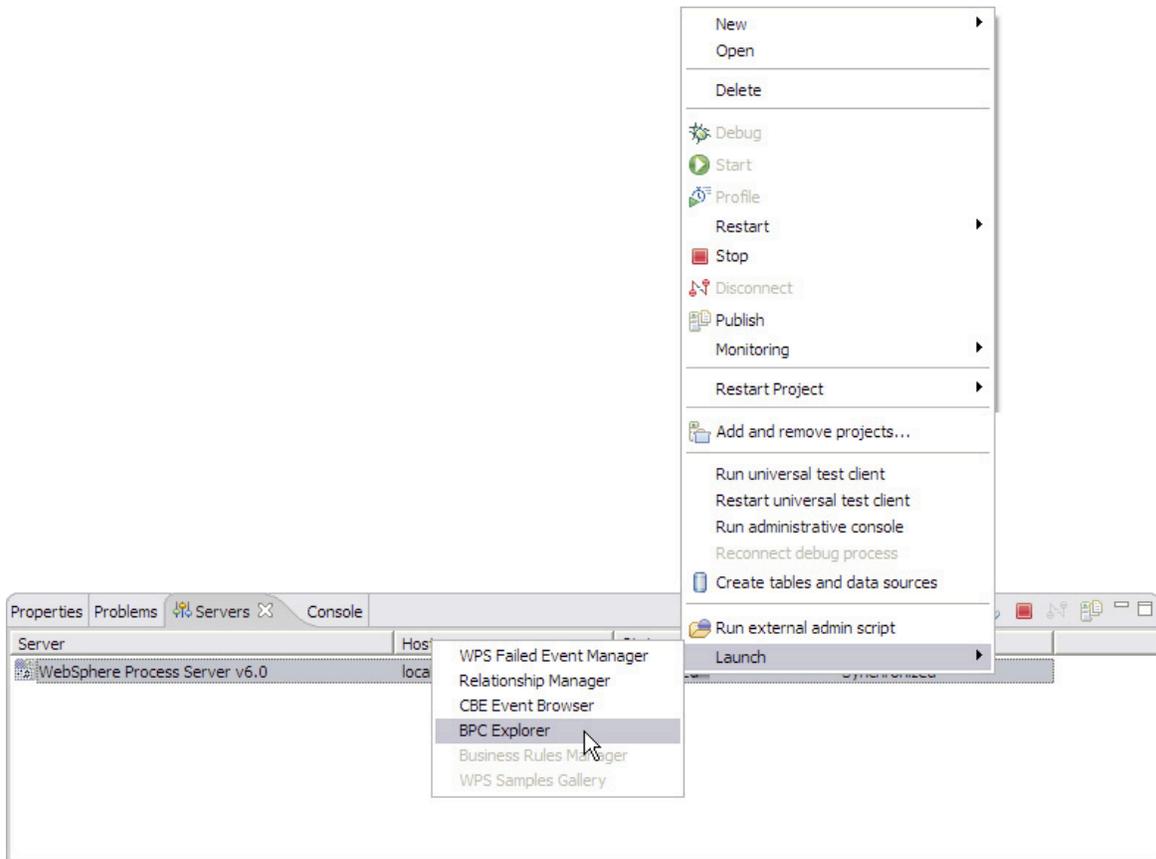
Su servidor de aplicaciones debería estar arrancado. Compruébelo en la pestaña de "Servers".



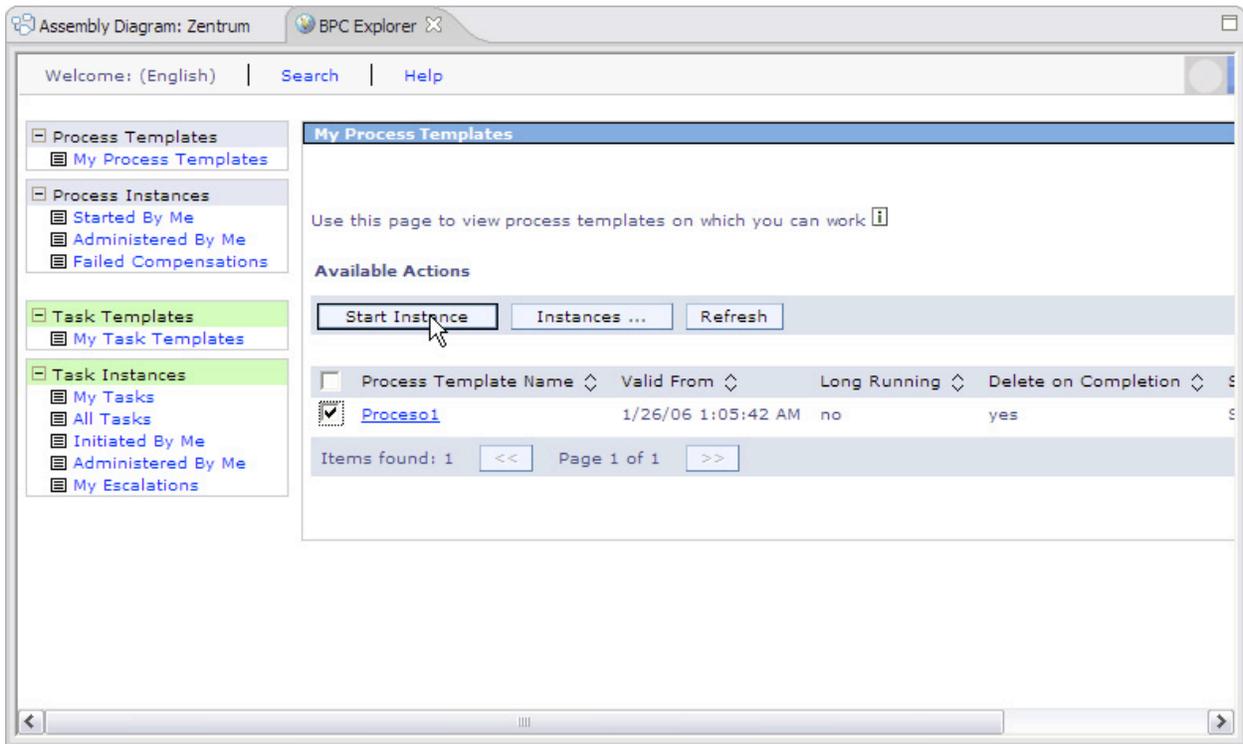
Asegúrese que los dos proyectos que hemos creado están instalados en el servidor.



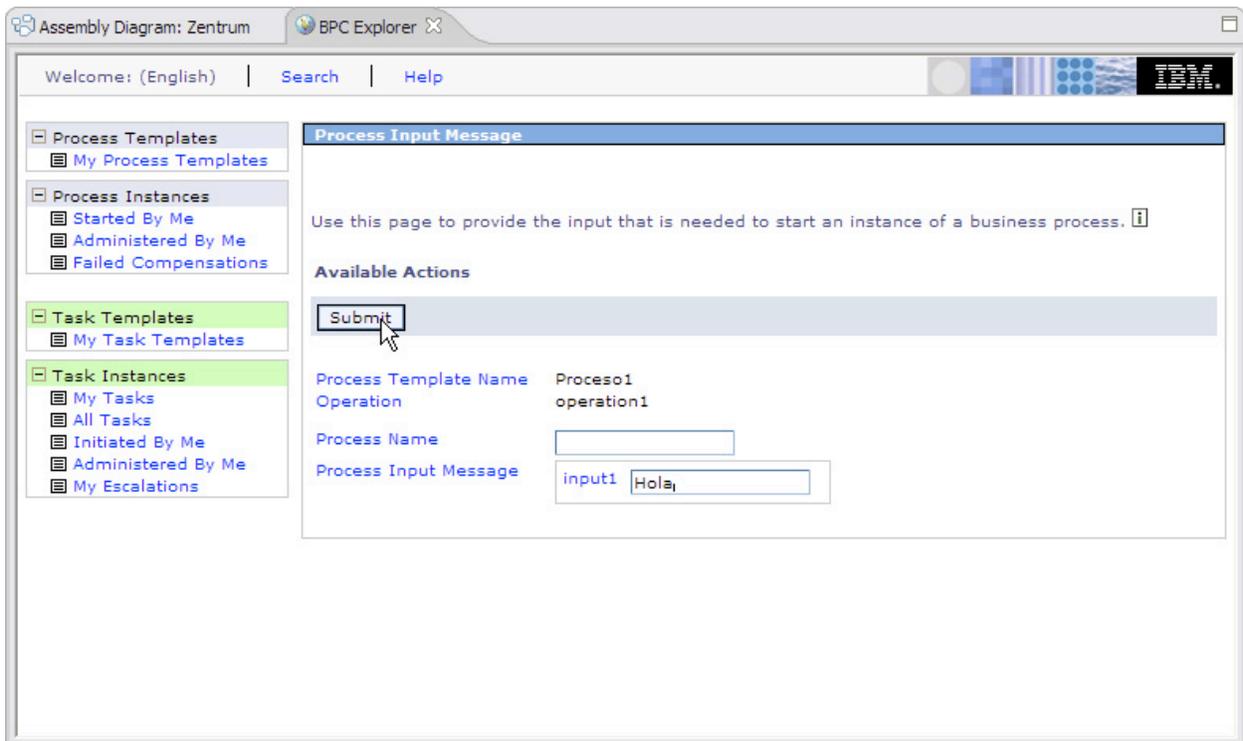
Arranque el Business Process Component (BPC) Explorer.



Haga click sobre “My Process Templates” para ver el proceso que acabamos de crear.



Seleccione el “Proceso1” y pulse el botón “Start Instance” para arrancar una instancia del proceso. Entre una cadena de prueba, por ejemplo “Hola” en la casilla “Input1”. Pulse submit para iniciar el proceso.



Si todo funciona correctamente, este es el resultado que debería obtener.

The screenshot shows the BPC Explorer interface. The left sidebar contains a tree view with the following categories and sub-items:

- Process Templates
  - My Process Templates
- Process Instances
  - Started By Me
  - Administered By Me
  - Failed Compensations
- Task Templates (highlighted)
  - My Task Templates
- Task Instances (highlighted)
  - My Tasks
  - All Tasks
  - Initiated By Me
  - Administered By Me
  - My Escalations

The main content area is titled "Process Output Message" and contains the following information:

Use this page to view the results of a business process that you started. ⓘ

Process Template Name	Proceso1
Operation	operation1
Process Input Message	input1 Hola
Process Output Message	output1 La cadena de entrada es corta

Pruebe con otros valores para asegurarse que el proceso funciona correctamente.