

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

Control de autenticación y
acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)



Empieza el jaleo: publicamos el libro
INFORMÁTICA PROFESIONAL
LAS REGLAS NO ESCRITAS PARA TRIUNFAR EN LA EMPRESA

-¿Crees que la informática es sólo programar?
-¿Sabrías organizar eficientemente un equipo?
-¿Quién te resuelve las dudas sobre esta profesión?..... o te siembra más?

Hosting patrocinado por **enredados**



[Inicio](#) [Quiénes somos](#) [Tutoriales](#) [Formación](#) [Comparador de salarios](#) [Comentar libro](#) [Charlas](#) [Más](#)

Estas en: [Inicio](#) [Tutoriales](#) Tutorial de BPEL con OpenESB (1)

Últimas Noticias

- » Impresiones y video de asistentes al primer coding-dojo de agilismo.es
- » Tarta y 7º aniversario de AdictosAlTrabajo.com
- » Formación por calendario
- » Comentando el libro: Eneagrama y éxito personal de Ginger Lapid-Bogda
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Disponible la primera versión de los plugins para integrar Maven y Bugzilla.
- » Lanzamiento del nuevo Web de Autentia
- » Si se pregunta ¿Qué ofrece este Web?
- » *Substitutos de Comida Analítica*

+Noticias Destacadas

- » Impresiones y video de asistentes al primer coding-dojo de agilismo.es
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Autentia patrocina el primer Agile Open Spain
- » Lanzamiento del nuevo Web de Autentia

+Comentarios Cómic

+Enlaces

Catálogo de servicios Autentia (PDF 6,2MB)



En formato comic...

Acceso de usuarios registrados:

E-mail:
Contraseña:

[Deseo registrarme](#)
[He olvidado mis datos de acceso](#)

Registra tu empresa:

Descubre las ventajas de registrar tu empresa en AdictosAlTrabajo...
[Registrar mi empresa](#)
[Listado de empresas ya registradas](#)



☐ Web
☒ www.adictosaltrabajo.com

Últimos tutoriales

2009-12-29
[Tutorial de BPEL con OpenESB \(I\)](#)

2009-12-28
[Pruebas funcionales de servicios web con soapUI](#)

2009-12-28
[SoapUI: jugando con web services](#)

2009-12-17
[¿Qué son el cloud computing y google app engine?](#)

2009-12-14
[JavaBean DataSource Ireport](#)

2009-12-11
[Contract-First web services con Visual Studio 2008](#)

2009-12-09
[Integrando Sonar con Hudson](#)

2009-12-09
[Apache + Tomcat: Balanceo de carga y alta disponibilidad](#)

2009-12-08
[MySQL: Replicación de bases de datos en MySQL](#)

2009-12-07
[Analizando la calidad del código Java con Sonar](#)

2009-12-03
[Instalar OpenESB 2.1 e Introducción](#)

2009-11-25
[Tutorial de Google Forms](#)

2009-11-23
[Alfresco - Modificando y eliminando contenido desde nuestras aplicaciones Java](#)

2009-11-23
[Alfresco - Añadiendo contenido desde nuestras aplicaciones Java](#)

2009-11-23
[Haciendo funcionar Google Chromium Operating System sobre Windows XP](#)

2009-11-23
[Redimensionar el tamaño de una partición de VirtualBox](#)

Tutorial desarrollado por



Iván García Puebla

Consultor tecnológico de desarrollo de proyectos informáticos.
Puedes encontrarme en [Autentia](#)
Somos expertos en Java/JEE

Catálogo de servicios de Autentia

[Descargar \(6,2 MB\)](#)

[Descargar en versión comic \(17 MB\)](#)

[AdictosAlTrabajo.com](#) es el Web de difusión de conocimiento de Autentia.



[Catálogo de cursos](#)

Descargar este documento en formato PDF: [introduccion-BPEL-openesb.pdf](#)

Fecha de creación del tutorial: 2009-12-29

Tutorial de BPEL con OpenESB (I)

Índice

1. [Tutorial de BPEL con OpenESB \(I\)](#)
 1. [Introducción](#)
 2. [El proceso de negocio](#)
 3. [Identificar los servicios web requeridos](#)
 4. [Modelar el proceso de negocio con BPEL](#)
 1. [Generar el descriptor del proceso BPEL](#)
 2. [Importar descriptores WSDL de los servicios involucrados](#)
 3. [Diseñar el proceso BPEL gráficamente](#)
2. [Tutorial de BPEL con OpenESB \(II\)](#)
 1. [Desplegar el módulo BPEL en el ESB](#)
 2. [Probar el proceso de compra](#)
 3. [Conclusión](#)

Introducción

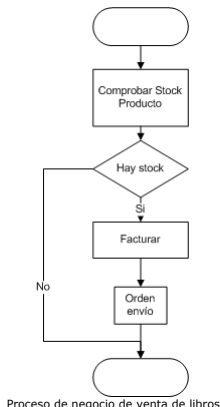
En este tutorial vamos a aprender a crear procesos **BPEL** practicando con un ejemplo: un proceso de negocio de venta online de libros. La empresa *TuLibroOnline* decidió adoptar SOA para sus infraestructuras IT y acualmente se encuentra en la fase de composición de servicios. Así pues, le llega el turno al proceso de venta de libros y se decide implementarlo con un proceso BPEL orquestando servicios web de las áreas implicadas en el negocio.

Para desarrollar el ejemplo es necesario tener instalado **OpenESB 2.1** tal como se muestra en el tutorial **OpenESB 2.1. Instalación e introducción al entorno**, y tener unos conocimientos básicos de NetBeans y GlassFish. El código fuente puede descargarse aquí: **BPEL-openESB_adictosaltrabajo.zip** y el proyecto de pruebas de soapUI aquí: **BPEL-openESB_adictosaltrabajo.zip**.

Hemos dividido el tutorial en dos partes debido a su envergadura. La segunda parte del tutorial se encuentra en **Tutorial de BPEL con OpenESB (II)**.

El proceso de negocio

Antes de programar nada, tenemos que comprender muy bien el proceso de negocio que pretendemos implementar con alguna tecnología. Para ello nos reunimos nuevamente con un responsable de negocio del área de ventas y tras comentarle los avances de la implantación tecnológica y lo que le permite ya hacer a la empresa, le pedimos valide el proceso de venta de libros. Finalmente se alcanza la propuesta:



Proceso de negocio de venta de libros

Los detalles del proceso son los siguientes:

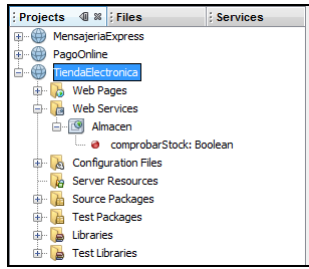
- La orden de venta de inicia aportando el ID de cliente registrado que realiza la compra, el ISBN del libro que adquiere, el número de unidades y el precio por unidad.
- Se comprueba en almacén que haya stock suficiente para el pedido. En su defecto el proceso finaliza y lo notifica, para así ofrecer alternativas al cliente y no perder la oportunidad de venta.
- Si hay unidades suficientes, hemos de invocar al sistema de pago online que tenemos contratado con nuestra entidad financiera. La entidad ya dispone de los datos de los clientes registrados, por lo que se encarga de hacer las tramitaciones necesarias y asegurar el pago.
- Finalmente emitimos la orden de envío a una empresa de mensajería. Facilitaremos un identificador de pedido para que lo recoja en almacén y lo reparta.
- El resultado del proceso será una confirmación de la venta realizada.

Identificar los servicios web requeridos

Como siguiente paso tenemos que identificar los servicios web que realizan las funcionalidades concretas de cada paso del proceso. En nuestro caso son:

- Servicio web del almacén:
 - `comprobarStock`, que espera el ISBN del libro y las unidades de compra, y responde si existe o no stock suficiente
- Servicio web del sistema de pago online:
 - `facturar`, con un `idCliente` y una cantidad. Devuelve un código de factura
- Servicio web de la empresa de mensajería:
 - `enviarProducto`, con nuestro identificador de la empresa y el pedido a servir. Nos responde con el tiempo en días estimado de reparto.

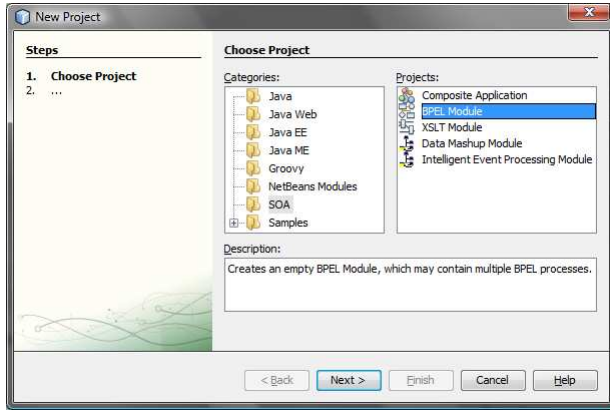
Estos servicios están implementados en los proyectos *MensajeríaExpress*, *PagoOnline* y *TiendaElectronica* del código fuente del tutorial. Deben ser abiertos como proyectos de NetBeans (menú **File** | **Open Project**):



Servicios Web del proceso de negocio en NetBeans

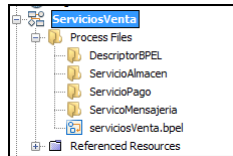
Modelar el proceso de negocio con BPEL

Creamos un nuevo proyecto BPEL en NetBeans: File | New Project | SOA | BPEL Module:



Crear un Módulo BPEL en OpenESB

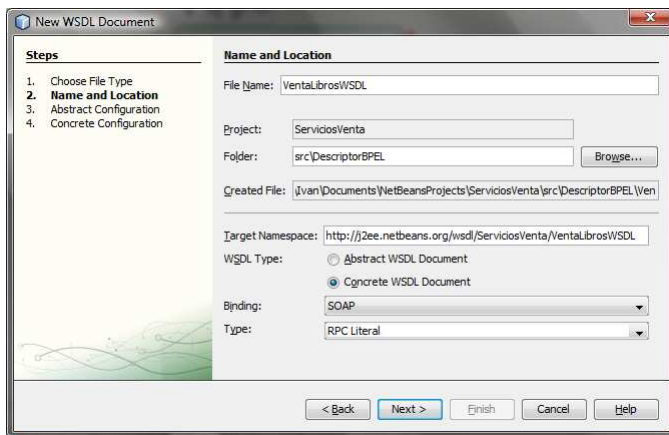
Le asignamos el nombre ServiciosVenta. Sobre la estructura inicial del proyecto creamos las carpetas: DescriptorBPEL, ServicioAlmacen, ServicioPago, ServicioMensajería (botón derecho sobre Process Files | New | Other | Folder e introducimos el nombre en el campo Folder Name):



Estructura inicial del módulo BPEL

Generar el descriptor del proceso BPEL

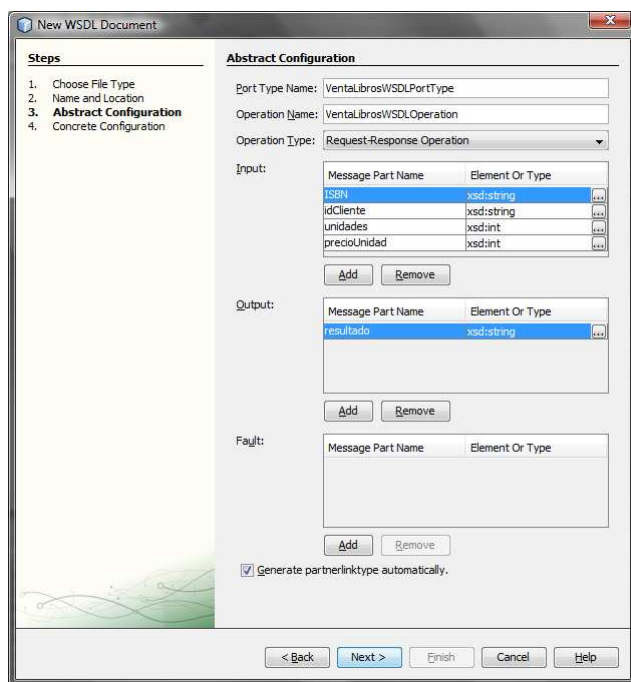
Un proceso BPEL se expone al resto como si de un servicio web se tratara. Por ello debe tener su propio descriptor WSDL. Lo creamos (botón derecho sobre DescriptorBPEL | New | WSDL Document) e introducimos los valores de la imagen:



Crear descriptor WSDL del proceso BPEL

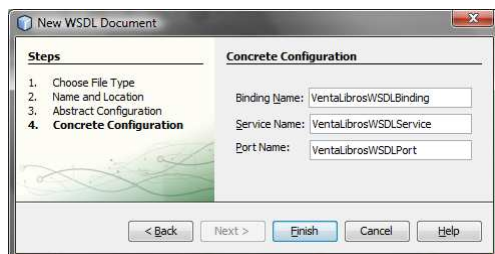
En el siguiente paso del asistente introducimos los siguientes valores:

2009-11-20	El Arte de las Presentaciones. Siguiendo la Senda Zen.
2009-11-18	Tutorial basico de google wave bots
2009-11-13	Introducción a Escritorios Animado (Winamp y MilkWave)
2009-11-12	Maven Assembly Plugin: empaquetando aplicaciones con Maven para la ejecución de procesos batch.
2009-11-12	Tutorial básico sobre Google Application Engine
2009-11-11	Instalación de Glassfish 2.1
2009-11-26	Rational Software Architect y Rational RequisitePro
2009-10-27	Cómo conseguir que Subversion avise a Hudson para lanzar una build
2009-10-26	Cómo instalar Hudson en Apache Tomcat
2009-10-26	Estilos externos en iReport
2009-10-14	Echo en MySQL
2009-09-28	iReport : Solución al problema de los parámetros dinámicos.
2009-09-27	Mavenizar Liferay SDK
2009-09-27	Crear un plugin con Liferay SDK
2009-09-23	EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Apache Open EJB.
2009-09-21	Primeros pasos con JBoss Seam
2009-09-21	Integración con sistemas de Bug Tracking desde NetBeans 6.7
2009-09-14	EJB 3.0 y pruebas de persistencia con Maven, JUnit 4 y Embedded JBoss sobre Java 6.
2009-09-12	Instalación de Liferay en Tomcat existente
2009-09-11	Release Bugzilla Maven Plugin
2009-09-11	Enlazar Bugzilla con MavenChangesPlugin
2009-09-08	Sobre las reglas de codificación o... ¿de dónde salen esos caracteres "raros"?
2009-08-28	Cómo hacer deploy del site de Maven en SourceForge
2009-08-26	Ordenación por cantidades en informe cruzado
2009-08-20	Selenium IDE-Incorporando while en los test
2009-08-14	Blender y JMonkeyEngine. Exportación de archivos Blender y uso de los mismos en JMonkeyEngine
2009-08-14	5º tutorial TNT Concept Versión 0.16.1 Gestión de informes, vacaciones y utilidades
2009-08-14	Joomla 1.5. Instalación y configuración
2009-08-13	



Definición de la operación del servicio y sus argumentos

En el último paso aceptamos los valores propuestos:



Configuración concreta del descriptor

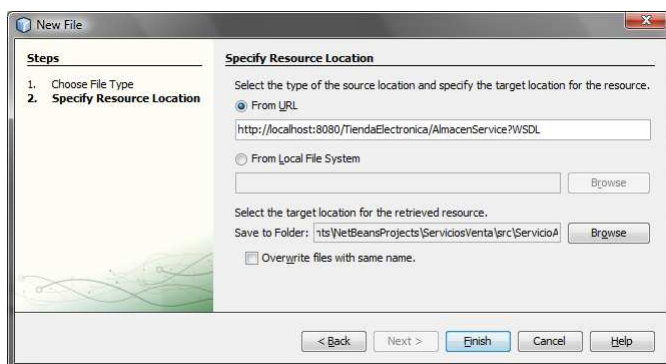
Finalizamos el asistente y ya tenemos definida la interfaz del proceso BPEL.

Importar descriptores WSDL de los servicios involucrados

Las aplicaciones web que hemos importado en NetBeans deberán estar desplegadas sobre el servidor GlassFish previamente arrancado. Así, podemos acceder a los contratos de los web services en las URL:

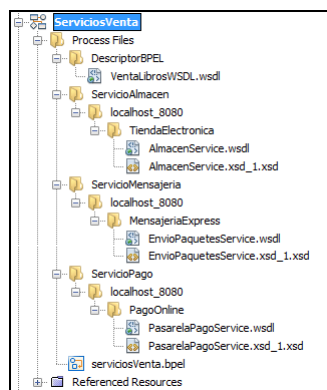
- <http://localhost:8080/MensajeriaExpress/EnvioPaquetesService?WSDL>
- <http://localhost:8080/PagoOnline/PasarelaPagoService?WSDL>
- <http://localhost:8080/TiendaElectronica/AlmacenService?WSDL>

Todo proceso BPEL necesitará los contratos WSDL de los servicios que utilice, para poder invocarlos. Para importar los WSDL, hacemos botón derecho sobre la carpeta `servicioAlmacen` | New | Other | XML | External WSDL Document(s) e introducimos la URL del descriptor del servicio de almacen:



Importamos los WSDL de los servicios utilizados

Repetimos la operación con las carpetas `ServicioPago`, `ServicioMensajeria` y sus correspondientes descriptores. El proyecto quedará por tanto:



Importamos los descriptores de los webservices utilizados

Cabe decir que si disponemos los WSDL en disco en vez de resolverlos de la web, también pueden ser importados de manera similar. Incluso tendríamos una estructura de carpetas más limpia que la generada por NetBeans.

[Introducción a los diagramas EPC \(Event-Driven Process Chain\)](#)

2009-08-10
[Blender. Animaciones avanzadas y renderización](#)

2009-08-10
[Gestión de Calidad, tablón y seguimiento en TNT Concept Versión 0.16.1](#)

2009-08-10
[Cómo hacer una página web](#)

2009-08-06
[Tips And Tricks JUnit Spring](#)

2009-08-03
[Instalación de VirtualBox PUEL](#)

Últimas ofertas de empleo

2009-07-31
[T. Información - Operador \(día / noche\) - BARCELONA.](#)

2009-06-25
[Atención a cliente - Call Center - BARCELONA.](#)

2009-06-19
[Otras - Ingeniería \(minas, puentes y puentes\) - VALENCIA.](#)

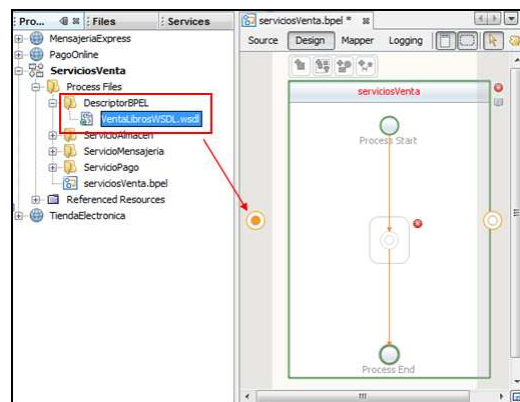
2009-06-17
[Comercial - Ventas - ALICANTE.](#)

2009-06-03
[Comercial - Ventas - VIZCAYA.](#)

[Anuncios Google](#)

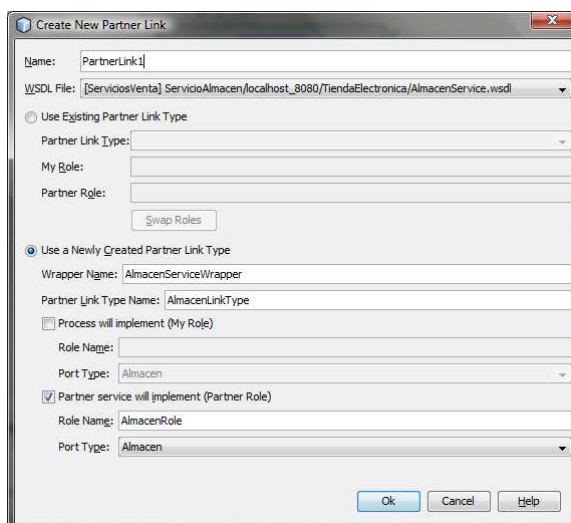
Diseñar el proceso BPEL gráficamente

Ahora viene lo más divertido :-). Hacemos doble click sobre `serviciosVenta.bpel` y accedemos al editor visual. Es intuitivo y se basa en arrastrar y soltar elementos en las zonas permitidas. En primer lugar insertamos el descriptor del proceso BPEL a la izquierda del diagrama:



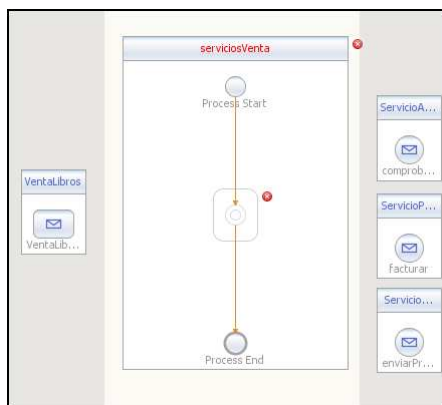
Insertar la interfaz WSDL del proceso BPEL

Pulsando sobre la representación del WSDL recién introducido y en su icono , accedemos a las propiedades. Cambiamos el nombre por defecto `PartnerLink1` por `VentaLibros` y aceptamos. De igual manera introducimos los descriptores de los webservices utilizados, pero esta vez a la derecha del diagrama, como proveedores de servicios. Al arrastrar y soltar, aparecerá el diálogo de creación de un nuevo enlace a un proveedor de servicio (en el cual aprovecharemos para cambiar el nombre `PartnerLink1` por el del servicio que corresponda, e.g. `ServicioAlmacen`):



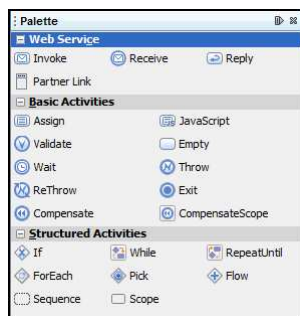
Crear enlace a servicio proveedor

El proceso quedará de la forma:



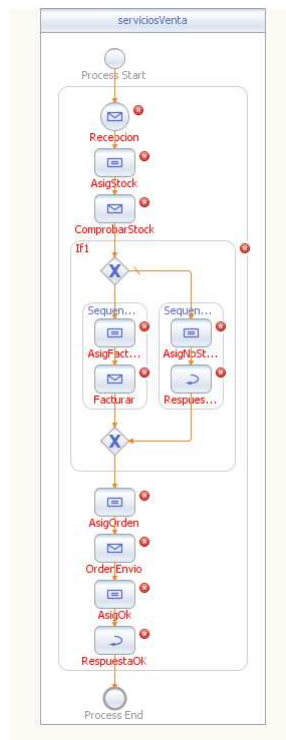
BPEL con los servicios proveedores (PartnerLinks)

A continuación utilizamos la paleta de actividades del editor BPEL de NetBeans:



Paleta de actividades del editor de BPEL

y componemos en unos minutos un proceso como el de la imagen siguiente (para cambiar los nombres de las actividades utilizamos su icono de edición ):



BPEL inicial con sus actividades

Comenzamos asociando las actividades de recepción, invocación y respuesta a los servicios. Hacemos doble click sobre la actividad de recepción e introducimos los valores que muestra la siguiente imagen. Para obtener la propiedad Input Variable, pulsamos sobre el botón Create... y aceptamos los valores por defecto, pues son significativos:

Editando la actividad de recepción

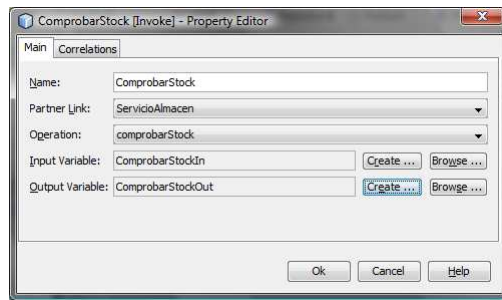
Tras aceptar, vemos en el BPEL que ha aparecido una flecha que une VentaLibros con la actividad de recepción. De igual manera hacemos con las actividades de respuesta. Editamos la última respuesta del proceso, RespuestaOk:

Editando la actividad de respuesta

Hacemos lo mismo con la actividad de respuesta contenida dentro de la estructura de decisión (RespuestaNoOk):

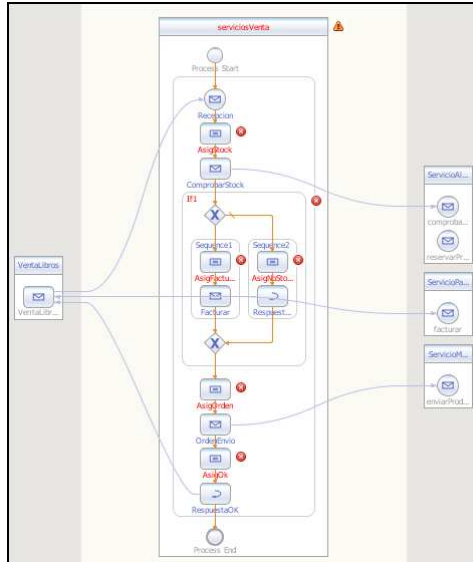
Editando la respuesta de falta de Stock

A continuación hacemos lo mismo para cada actividad de invocación. Comenzando por ComprobarStock:



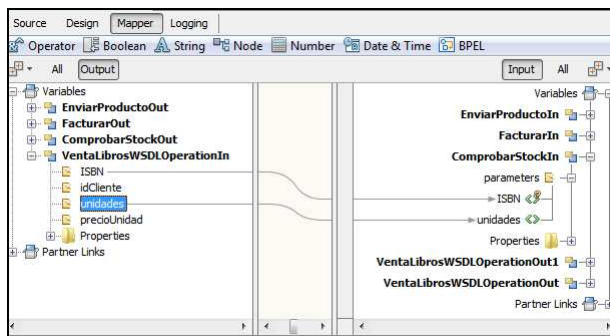
Invocación del servicio de stock

Repetimos la operación para el resto de invocaciones, y el proceso BPEL habrá tomado una forma más reconocible::



BPEL con las invocaciones a los servicios

A continuación editamos las variables de asignación. Como su nombre indica, cada una asigna variables a la actividad que le sigue. Por ejemplo, a la actividad de invocación AsignarStock espera unos valores de entrada (los del web service al que invoca). Estos valores los tenemos que asignar de la actividad que los tenga, en este caso la de recepción. Para ello pulsamos doble click sobre la actividad AsignarStock y se abre el siguiente mapeador de valores, al que hemos asociado relaciones entre partes de mensajes, pulsando en el nombre de variable origen y arrastrando la fecha hasta el destino:

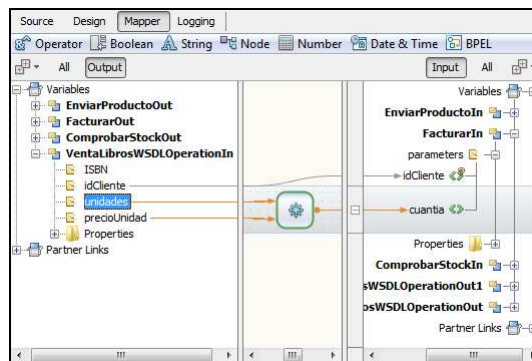


Asignación de valores entre partes de mensajes

A la izquierda hemos filtrado por valores de salida (Output) y a la derecha por variables receptoras (Input). En este caso estamos asignando los valores de la actividad de recepción (cuyo nombre de variable 'VentaLibrosWSDLOperationIn' fue definido en un paso anterior), a la variable de entrada de la invocación al web service de comprobación de Stock (ComprobarStockIn).

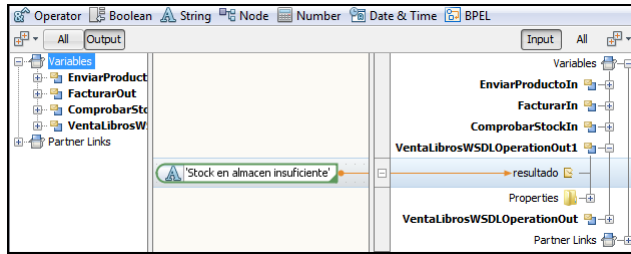
Para salir del mapeador y regresar el diagrama BPEL, hay que pulsar sobre la palabra Design en la parte superior del mapeador.

Editamos AsigFactura. En este caso El web service de facturar va a necesitar una variable cantidad, que es el resultado de multiplicar las unidades encargadas por el precio unitario. Para hacer esta transformación, seleccionamos en la parte derecha la variable cantidad. Cuando se ilumine la franja horizontal, desplegamos el menú operador y seleccionamos Multiplication. La salida de la multiplicación la asignamos a cantidad, y los operadores de entrada son unidades y precioUnidad. Arrastramos sus nombres hacia el operador, y quedará como en la siguiente imagen:



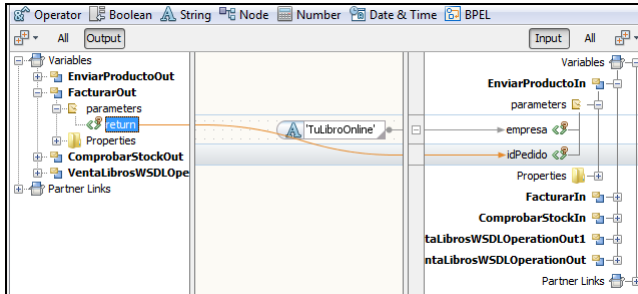
Asignar valores a la invocación del servicio de facturación

Editamos ahora AsignoStock. Estamos en el camino del 'else', y tenemos que asignar a la variable de la actividad de respuesta un valor que exprese que no se ha podido completar el proceso debido a la falta de stock de producto. Dicha variable es VentaLibrosWSDLOperationOut1 y su mensaje es resultado. Seleccionamos esta variable, accedemos al menú String, seleccionamos String Literal, introducimos la cadena Stock en almacen insuficiente y lo asignamos a resultado:



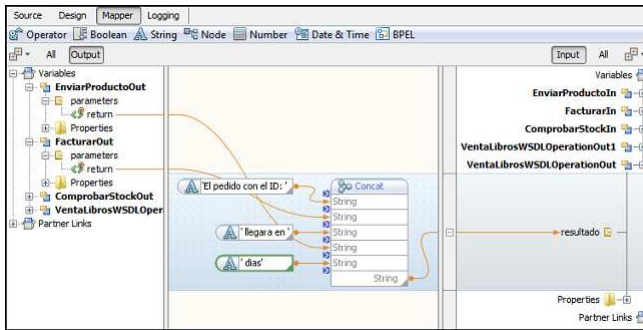
Respuesta indicando la falta de stock del producto

Tras la sentencia de selección nos encontramos con la actividad de asignación `AsigOrden`, que establecerá los valores de la variable `EnviarProductoIn` para la invocación al servicio de mensajería para que venga al almacén a recoger y repartir el pedido. `EnviarProductoIn` tiene dos parámetros en el mensaje: empresa, que será un literal, e `idPedido`, que será el resultado de la invocación al servicio de facturación. Por tanto la asociación quedará de la forma:



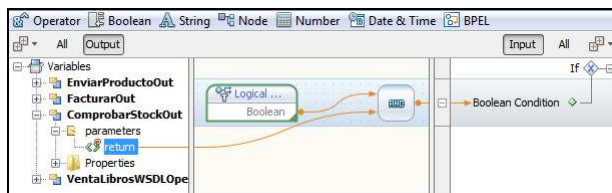
Valores para invocar al servicio de mensajería

Finalmente editamos `AsigOk`, que asignará a la actividad de respuesta final un mensaje de confirmación de venta indicando el id de albarán dado por el servicio de facturación y los días que tardará en entregarse, indicado en la respuesta del servicio de mensajería. Concatenaremos en este caso varios literales con la respuestas de los servicios:



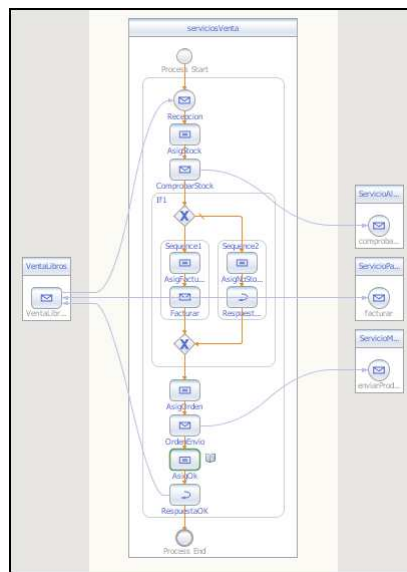
Asignación de resultado a la actividad de respuesta

Por último tenemos que establecer la condición de la estructura de bifurcación. Pulsamos dos veces sobre la estructura del `If`, y utilizando el menú `Boolean` y los elementos `Logical And` y `Logical True` componemos la condición booleana en base a la respuesta del servicio de comprobación de stock:



Condición del proceso sobre la existencia de stock

En este punto el proceso BPEL estará finalizado, sin mensajes de error ni advertencias, listo para ser probado:



Aspecto final del proceso de negocio con BPEL

Continuación...

Este tutorial continua en [Tutorial de BPEL con OpenESB \(II\)](#)

¿Qué te ha parecido el tutorial? Déjanos saber tu opinión y ivota!

Muy malo Malo Regular Bueno Muy bueno

Votar

(Sólo para usuarios registrados)

» [Regístrate y accede a esta y otras ventajas](#) «

Anímate y coméntanos lo que pienses sobre este tutorial

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Enviar comentario

(Sólo para usuarios registrados)

» [Regístrate y accede a esta y otras ventajas](#) «

Autor

Mensaje de usuario registrado

- Puedes inscribirte en nuestro servicio de notificaciones [haciendo clic aquí](#).
- Puedes firmar en nuestro libro de visitas [haciendo clic aquí](#).
- Puedes asociarte al grupo AdictosAlTrabajo en XING [haciendo clic aquí](#).
- [Añadir a favoritos Technorati](#).



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
Tutorial de BPEL con OpenESB (I)	En este tutorial vamos a aprender a crear procesos BPEL practicando con un ejemplo: un proceso de negocio de venta online de libros.	2009-12-29	13	-	-	
SoapUI: jugando con web services	SoapUI es una aplicación muy versátil que nos permite probar, simular y generar código de servicios web de forma ágil. En este tutorial introducimos esta herramienta, mostrando su instalación y un primer ejemplo de uso con un web service público	2009-12-28	107	-	-	
Pruebas funcionales de servicios web con soapUI	Las pruebas unitarias en cualquier paradigma de programación son, más que una buena práctica, una garantía para obtener un software robusto y (más) fácilmente mantenible. Como responsables de diseño o desarrollo de web services hemos de aplicar estas buen	2009-12-28	91	-	-	
JavaBean DataSource Ireport	La particularidad del caso que nos ocupa, es conseguir que la fuente de datos del informe sea una lista de JavaBeans y no una consulta definida previamente en el informe.	2009-12-14	485	-	-	
Contract-First web services con Visual Studio 2008	Utilizaremos Visual Studio 2008 con el addin WSCF.blue para generar contract-first web services (servicios web dirigidos por contrato)	2009-12-11	394	-	-	
Instalar OpenESB 2.1 e Introducción	En este tutorial veremos como descargar e instalar OpenESB y explicaremos sus funcionalidades	2009-12-03	733	-	-	
Tutorial de Google Forms	En este breve tutorial vamos a ver cómo podemos usar la herramienta Google Forms, para crear rápida y cómodamente un formulario de recogida de datos online (y de paso opinar sobre el Real Madrid).	2009-11-25	990	-	-	
Tutorial básico sobre Google Application Engine	En este tutorial veremos cómo ejecutar una simple aplicación web en la infraestructura de Google	2009-11-12	1200	Bueno	1	
Instalación de Glassfish 2.1	En este tutorial nos veremos cómo instalar el servidor de aplicaciones GlassFish. Además veremos los primeros pasos, como entrar en la consola de administración del servidor, y desplegar una aplicación EAR (Enterprise Application)	2009-11-11	1541	Bueno	1	
EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Apache Open EJB.	Continuamos buscando una buena solución para llevar a cabo test de EJBs, ahora con OpenEJB.	2009-09-23	1906	-	-	

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.