MTIS	Metodologías y Tecnologías de Integración de Sistemas
	Práctica 3
	BPEL: Ejercicio Guiado 6 Tipos de datos complejos

Ejercicio Guiado 6 Tipos de datos complejos

Prerrequisitos

1. Instalación del contenedor Web Apache Tomcat 8.0.20
2. Instalación de MySQL 5.x
3. Instalación de Apache ODE 1.3.6
4. Instalación de Eclipse Java EE Mars
5. Máquina virtual de Java 1.8
6. Apache Axis2

Objetivos

El objetivo de estos ejercicios es familiarizarse con el estándar WS-BPEL y sus principales elementos para la composición. Para ello usaremos como entorno de modelado de procesos de Eclipse BPEL Designer y Apache ODE como motor de ejecución de procesos.

1. Enunciado

En este ejercicio se introducirá al alumno a los siguientes elementos de BPEL:

- *Tipo de datos complejos*

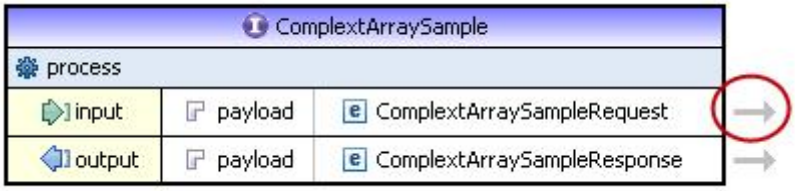
En este ejercicio partiremos de lo aprendido en los ejercicios anteriores de BPEL. Las partes comunes no serán descritas en el documento y sólo se hará referencia a la introducción de las nuevas actividades.

Vamos a ver cómo crear los tipos de datos complejos y cómo manejarlos dentro de un proceso BPEL.

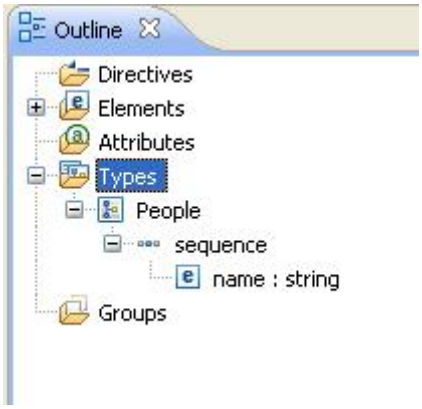
1.1 Crear un nuevo proyecto BPEL llamado "*BPEL_ComplexArray*" , seleccionando en el menú *File → New → Other… → BPEL 2.0 → BPEL Project* y seleccionar *Next.*

1.2 Crear un archivo BPEL (New->Others->BPEL 2.0->New BPEL Process File) denominado *ComplextArraySample.bpel* , con el espacio de nombres *http://complexarray.bpel.mtis* de tipo síncrono.

1.3 Abre el fichero *ComplextArraySampleArtifacts*.wsdl y pulsa en *ComplexArraySampleRequest* para abrir el *Inline Schema* de esa variable:

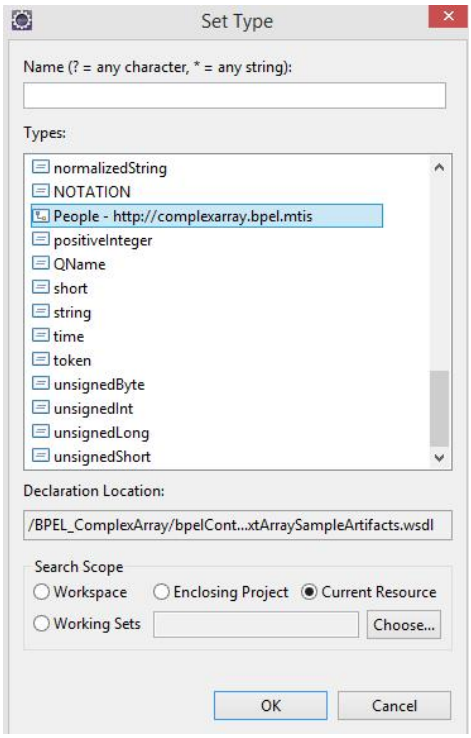


1.4 En la vista de outline, pulsa el botón derecho y Add Complex Type, cámbiale el nombre a *people*. Pulsa el botón derecho en *people*, selecciona *Add Elements* para crear *name* con tipo *string*.

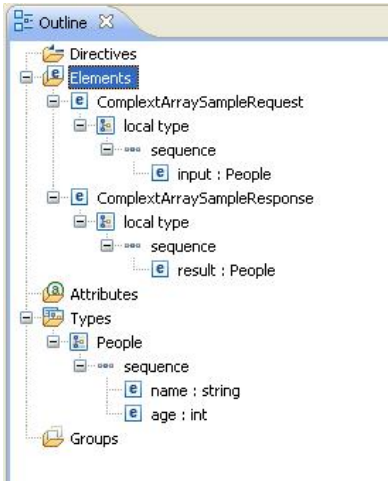


1.5 Haz lo mismo para *age* de tipo *int*.

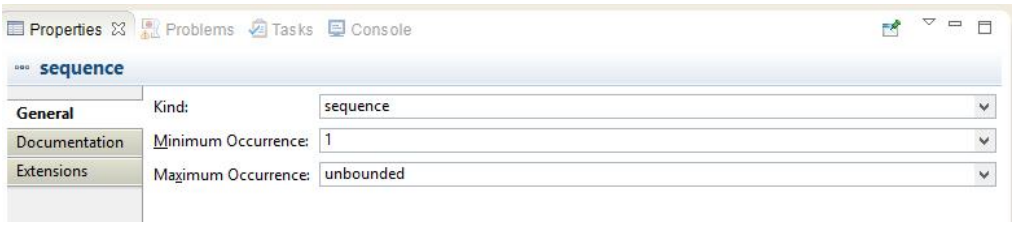
1.6 En la vista *Outline*, selecciona *Elements->ComplexArraySampleRequest->local type->sequence->input*, entonces en la vista propiedades en *Type* selecciona Browse y selecciona *People*.



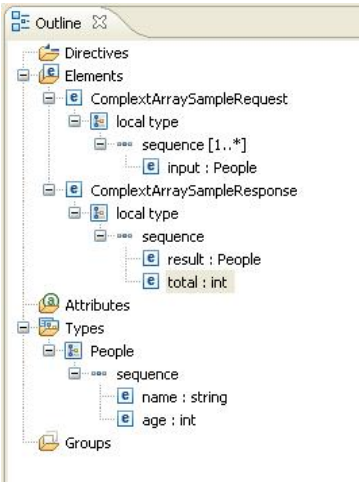
1.7 De igual forma, cambiar el tipo a *Elements->ComplexArraySampleResponse->local type->sequence->result* de *string* a *People .*



1.8 Los parámetros de entrada, van a ser un array de People y en el resultado, tendremos también el número de elementos de dicho array. Pulsa en *Elements->ComplexArraySampleRequest->local type->sequence*, en la vista de propiedades, selecciona *Minimum Occurrence* a 1 y *Maximum Occurrence* como *unbounded.*

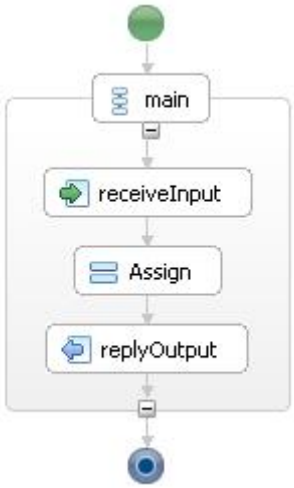


1.9 Botón derecho en *Elements->ComplexArraySampleResponse->local type->sequence*, selecciona *Add Element*, llámalo *total* e indícale de tipo *int*.



1.10 Guarda los cambios y ahora procedemos al diseño con BPEL.

1.11 Abre *ComplextArraySample.bpel* y arrastra una acción *assign* entre *receiveInput* y *replyOutput*.



1.12 Dentro de los detalles del *assign*, asigna la expresión *count($input.payload/tns:input)* a la variable *output->payload->total*.

1.13 Asigna también la expresión *$input.payload/tns:input[last()]* a la variable *output->payload->result.*

1.14 Guardamos los cambios y procedemos al despliegue como en los casos anteriores.

1.15 Finalmente comprobamos los resultados mediante SOAPUI.