



**UNIVERSIDAD DEL AZUAY**  
**FACULTAD DE ADMINISTRACIÓN**  
**ESCUELA DE INGENIERIA DE SISTEMAS**

**“Orquestación de servicios web aplicado a una solicitud de crédito comercial  
utilizando la herramienta Oracle BPEL Process Manager.”**

**Tesis previa a la obtención del título de ingeniero de sistemas**

**Autor: Luis Miguel Arévalo Ajila**

**Director: Ing. Marcos Orellana**

**CUENCA, ECUADOR**

**2013**

## **DEDICATORIA**

Esta tesis dedico a toda mi familia, ya que con su apoyo incondicional pude llegar a culminar mi carrera. A mis padres y hermanos por su apoyo y confianza que han puesto en mí día tras día. A mi madre por sus consejos, cuidados y amor que sirvieron para seguir adelante. A mi padre por aconsejarme siempre y por darme el apoyo y recursos necesarios. A mis hermanos, abuelos, tíos y primos que siempre estuvieron pendientes de mí.

## **AGRADECIMIENTOS**

Agradezco a Dios por darme la vida y el conocimiento para seguir cumpliendo todos mis sueños. A todos mis profesores de la Universidad del Azuay por darme sus consejos y transmitirme sus enseñanzas para ser mejor persona y profesional en la vida. A mi director de tesis el Ing. Marcos Orellana por su guía y apoyo en este proyecto de titulación.

# Índice de Contenidos

<b>Dedicatoria.....</b>	<b>ii</b>
<b>Agradecimientos.....</b>	<b>iii</b>
<b>Índice de Contenidos.....</b>	<b>iv</b>
<b>Índice de Ilustraciones y cuadros.....</b>	<b>vii</b>
<b>Resumen.....</b>	<b>viii</b>
<b>Abstract.....</b>	<b>xv</b>

## **CAPITULO 1**

### **CONCEPTOS BASICOS**

1.1 Arquitectura Orientada a Servicios (SOA) .....	1
1.1.1 Introducción.....	1
1.1.2 Capas y Componentes SOA .....	2
1.2 Oracle BPEL Process Manager .....	8
1.2.1 Introducción.....	8
1.2.2 Características .....	9
1.2.3 Arquitectura.....	10
1.2.4 Servicios de Integración .....	12
1.3 Web Service .....	14
1.3.1 Introducción.....	14
1.3.2 Estándares Relacionados .....	14
1.3.3 Integración en los Procesos de Negocio.....	15
1.3.4 Composición de servicios web .....	16

## **CAPITULO 2**

### **INTEGRACION CON ORACLE BPEL PROCESS MANAGER Y WEB SERVICES**

2.1 Orquestación de Servicios.....	20
2.1.1 Introducción.....	20
2.1.2 Estructuras de un Servicio BPEL .....	21
2.1.3 Elementos Básicos.....	22
2.1.4 Actividades Básicas.....	26
2.1.5 Invocación de Servicios.....	32

2.1.6	Gestión de la comunicación en los Procesos BPEL .....	34
2.2	Adaptadores: Tecnología y Aplicaciones.....	37
2.2.1	Introducción.....	37
2.2.2	Tecnologías Relacionadas .....	37
2.2.3	Aplicaciones Relacionadas .....	38
2.2.4	Tipos de Servicios de Adaptadores .....	39
2.3	Flujos de Trabajo Humano.....	41
2.3.1	Introducción.....	41
2.3.2	Especificación de Requerimientos de Flujo de Trabajo Humano .....	42
2.3.3	Gestión de los Servicios en el Flujo de Trabajo de Oracle BPEL .....	43

### **CAPITULO 3**

#### **ANALISIS Y DISEÑO**

3.1	Definición General del Sistema .....	45
3.2	Análisis de Requerimientos de solicitudes de Crédito.....	45
3.2.1	Diagramas de Casos de Uso de solicitudes de Crédito.....	46
3.2.2	Diagrama de Clases de las solicitudes de Crédito .....	52
3.3	Modelo Conceptual de solicitudes de Crédito .....	53
3.4	Diagrama de Componentes de solicitudes de Crédito.....	54
3.5	Diseño de la Arquitectura para el Sistema de Solicitudes de Crédito.....	56
3.6	Procesos de Interacción del Sistema de solicitudes de Crédito.....	57
3.7	Diseño de la Solución.....	59
3.7.1	Plataforma Tecnológica .....	59
3.7.2	Diseño de las Interfaces de solicitudes de Crédito .....	61
3.7.3	Diseño de las Interfaces Básicas en PHP y .Net .....	62
3.7.4	Diseño de la Base de Datos de solicitudes de Crédito.....	64

### **CAPITULO 4**

#### **DESARROLLO**

4.1	Descripción del Ambiente de Desarrollo .....	66
4.2	Descripción de las Herramientas para el Desarrollo .....	67
4.2.1	Definición de las Herramientas para Desarrollo de Orquestación.....	67

4.2.2	Definición de las Herramientas de Bases de Datos .....	68
4.2.3	Definición de las Herramientas de Programación .....	68
4.3	Configuraciones .....	69
4.3.1	Configuración de las Herramientas de Bases de Datos .....	69
4.3.2	Configuración de las Herramientas de Orquestación .....	71
4.3.3	Configuración de las Herramientas de Programación .....	83
4.5	Procedimientos para el desarrollo del Sistema de Solicitud de Crédito.....	97
4.5.1	Procedimientos Preliminares .....	97
4.5.2	Orquestación de Servicios Web.....	101
4.5.3	Servicios de Transformación en BPEL.....	108
4.6	Invocación y Accesos a un Servicio BPEL a través de Adaptadores .....	110
4.7	Procesamiento de los Servicios de Notificación .....	112

## **CAPITULO 5**

### **VALIDACIONES Y PRUEBAS**

5.1	Análisis y Registro de una Solicitud de Crédito Comercial.....	114
5.1.1	Análisis de una Solicitud de Crédito.....	114
5.1.2	Verificación de la Información .....	120
5.1.3	Creación de la Información.....	121
5.2	Evaluación de Despliegue .....	123
5.3	Evaluación de la Consola de Oracle BPEL.....	126
5.4	Auditoria de Ejecución.....	127
5.5	Verificación del Funcionamiento de la Aplicación.....	128
5.6	Verificación de Interacción con la Base de Datos .....	129
5.7	Comprobación del Funcionamiento de Adaptadores de Bases Externas .	130
5.8	Prueba del Servicio de Notificación.....	131

## **CAPITULO 6**

### **CONCLUSIONES Y RECOMENDACIONES**

6.1	Conclusiones .....	133
6.2	Recomendaciones.....	134
	BIBLIOGRAFÍA .....	136

## Índice de Ilustraciones y cuadros

Figura 1.1: Arquitectura Orientada a Servicios

Figura 1.2: Capas de una SOA

Figura 1.3: Arquitectura de Servicios Web

Figura 1.4: Ciclo de Vida de SOA

Figura 1.5: Oracle BPEL Process Manager

Figura 1.6: Arquitectura de Oracle BPEL Process Manager

Figura 1.7: Integración de Oracle en los procesos de negocio

Figura 1.8: Diagrama de pila de Estándares Relacionados.

Figura 1.9: Relación entre los elementos de WSDL

Figura 1.10: Relación entre tecnología y negocios en UDDI

Figura 2.1 Orquestación de Servicios.

Figura 2.2 Estructura de un Servicio BPEL

Figura 2.3 Relación Entre Partners, Partner Links y Partner Link Types

Figura 2.4 Diálogo PartnerLink

Figura 2.5 Diálogo Variable

Figura 2.6 Diálogo Receive

Figura 2.7 Diálogo Receive

Figura 2.8 Diálogo Invoke

Figura 2.9 Diálogo Assign

Figura 2.10 Diálogo Throw

Figura 2.11 Diálogo Wait

Figura 2.12 Diálogo Empty

Figura 2.13 Invocación de un Servicio Síncrono

Figura 2.14 Invocación de un Servicio Asíncrono

Figura 2.15 Comunicación de los Servicios BPEL

Figura 2.16 Servicios del Bus de Aplicaciones de Oracle

Figura 2.17 Servicios de Adaptadores

Figura 2.18 Human Workflow

Figura 3.1 Aprobación de crédito

Tabla 3.1 Registro Civil

Tabla 3.2 Registro de Propiedad

Tabla 3.3 Información del Cliente

Tabla 3.4 Solicitud Comercial

Figura 3.2 Generación de Venta

Tabla 3.5 Genera Ventas

Tabla 3.6 Ingresar Cliente

Tabla 3.7 Consulta Cliente

Tabla 3.8 Modifica Cliente

Tabla 3.9 Elimina Cliente

Figura 3.3 Registro Civil

Tabla 3.10 Registra Persona

Figura 3.4 Registra Propiedad



Tabla 3.11 Registra Propiedad

Tabla 3.12 Modifica Propiedad

Figura 3.5 Diagrama Entidad-Relación Solicitudes de Crédito

Figura 3.6 Diagrama Conceptual Solicitud de Crédito

Figura 3.7 Diagrama de Componentes Solicitudes de Crédito

Figura 3.8 Diagrama de Componentes Administración de Personas

Figura 3.9 Diagrama de Componentes Administración de Propiedad

Figura 3.10 Arquitectura Solicitud de Crédito Comercial

Figura 3.11 Procesos de Interacción de una Solicitud de Crédito

Figura 3.12 Interfaz Lista de Solicitudes de Crédito

Figura 3.13 Interfaz Ingreso Solicitud de Crédito

Figura 3.14 Interfaz Registro de la Propiedad

Figura 3.15 Interfaz Registro Civil

Figura 3.16 Estructura de Tablas de Datos de Solicitud de Crédito

Figura 3.17 Estructura de Tablas de Datos del Registro de la Propiedad

Figura 3.18 Estructura de Tabla de Datos del Registro Civil

Figura 4.1 Herramientas de Diseño y Programación

Figura 4.2 Configuración de la Base de Datos

Figura 4.3 Sesión de SQLPLUS

Figura 4.4 Parámetros a Modificar

Figura 4.5 Parámetros Modificados

Figura 4.6 Inicio de Instalación

Figura 4.7 Selección del directorio de Instalación

Figura 4.8 Registro de Seguridad

Figura 4.9 Selección de Tipo de Instalación

Figura 4.10 Selección de Ruta de instalación

Figura 4.11 Selección de Método Abreviado

Figura 4.12 Resumen de Instalación

Figura 4.13 Configurar Propiedades del Sistema

Figura 4.14 Inserción de Variable del Sistema

Figura 4.15 Cuadro de Bienvenida de RCU

Figura 4.16 Creación o Eliminación de Repositorio

Figura 4.17 Conexión a la base de datos

Figura 4.18 Estado de Configuración de Conexión

Figura 4.19 Estructura de Selección de Componentes

Figura 4.20 Contraseñas de Esquema

Figura 4.21 Asignación de Tablespace

Figura 4.22 Detalles de Terminación

Figura 4.23 Ingreso de ubicación del JDK

Figura 4.24 Comprobación de Requisitos

Figura 4.25 Ubicación de Instalación

Figura 4.26 Selección del Servidor de Aplicaciones

Figura 4.27 Detalle de Instalación

Figura 4.28 Creación del Dominio SOA

Figura 4.29 Selección de Dominios

Figura 4.30 Especificación del Nombre y Ubicación del Dominio

Figura 4.31 Ingreso de Usuario y Contraseña del dominio

Figura 4.32 Configuración de esquema de componentes JDBC

Figura 4.33 Instalación Oracle JDeveloper

Figura 4.34 Selección del directorio raíz de Middleware

Figura 4.35 Finalización de Instalación

Figura 4.36 Ejecución de JDeveloper Studio 11.1.1.5.0

Figura 4.37 Selección del Rol

Figura 4.38 Selección del menú de actualización

Figura 4.39 Selección de tipo de actualización

Figura 4.40 Selección de componente de actualización

Figura 4.41 Resumen de Actualización SOA

Figura 4.42 Selección del archivo de configuración

Figura 4.43 Configuración de argumentos

Figura 4.44 Inicio de sesión a la consola Weblogic Server

Figura 4.45 Configuración de la consola de Weblogic Server

Figura 4.46 Selección del dominio de configuración

Figura 4.47 Personalizar Tabla

Figura 4.48 Configuración de Despliegue

Figura 4.49 Consola de Administración Empresarial

Figura 4.50 Selección de Propiedades Comunes

Figura 4.51 Selección de nivel de Auditora

Figura 4.52 IDE Oracle JDeveloper 11g

Figura 4.52 Detalles de la Pestaña Principal

Figura 4.53 Pestaña de Información Financiera

Figura 4.54 Pestaña de Referencias Datos del Trabajo

Figura 4.55 Pestaña de Referencias Personales

Figura 4.56 Ventana de Consulta de Clientes

Figura 4.57 Información detallada del Cliente

Figura 4.58 Página de Simulación de un Registro Civil

Figura 4.59 Página de Simulación de un Registro de Propiedad

Figura 4.60 Configuración de Conexión con la base de datos

Figura 4.61 Estructura de Dominio

Figura 4.62 Configuración del Pool de Conexiones Salientes

Figura 4.63 Ingreso de la ruta del JNDI

Figura 4.64 Lista de Conexiones Salientes

Figura 4.65 Configuración del Valor de Propiedad

Figura 4.66 Creación del Esquema XSD

Figura 4.67 Creación de un DatabaseAdapter

Figura 4.68 Creación de la Conexión con MySQL

Figura 4.69 Configuración de los parámetros de Conexión de MySQL

Figura 4.70 Selección de operación para la base de datos

Figura 4.71 Definición del criterio de Selección

Figura 4.72 Configuración de Bpel Process Manager

Figura 4.73 Enlace de Orquestación de Servicios

Figura 4.74 Estructura de una orquestación de Servicios web

Figura 4.75 Manipulación de datos con la actividad Assign

Figura 4.76 Copia de datos a una variable de salida

Figura4.77 Asignación de datos entre variables

Figura4.78 Copia de datos entre variables de dos Base de Datos

Figura4.79 Asignación de datos entre variables de salida de dos Base de Datos

Figura 4.80 Invocación a un adaptador de Base de Datos

Figura 5.1 Proceso de consulta de Información del Cliente

Figura 5.2 Mapeo de Variables

Figura 5.3 Manipulación de Variables de entrada

Figura 5.4 Invocación a la base del Registro Civil

Figura 5.5 Verificación de la Información del Registro Civil

Figura 5.6 Invocación a la base del Registro de Propiedad

Figura 5.7 Verificación de la Información del Registro de Propiedad

Figura 5.8 Estados de Decisiones

Figura 5.9 Orquestación de para la Actualización de Datos

Figura 5.10 Despliegue de Procesos

Figura 5.11 Acción de Despliegue

Figura 5.12 Configuración de Despliegue

Figura 5.13 Selección del Servidor

Figura 5.14 Selección del Dominio SOA

Figura 5.15 Resumen de Despliegue

Figura 5.16 Consola de evolución de procesos

Figura 5.17 Pruebas de Evaluación

Figura 5.18 Respuesta de Evaluación

Figura 5.19 Auditoria de Ejecución de Procesos

Figura 5.20 Resultados de Interacción de los procesos

Figura 5.21 Página de Flujo de tareas

Figura 5.22 Verificación de la información Orquestada

Figura 5.23 Verificación de la Información en la base de datos Oracle

Figura 5.24 Verificación de la Información en la base de datos SQL Server

Figura 5.25 Verificación de la Información en la base de datos MySQL

Figura 5.26 Ventana de Aprobación o Rechazo de una Solicitud de Crédito

Figura 5.27 Registro de una solicitud Aprobada

## **Resumen**

La Arquitectura Orientada a Servicios (SOA) es un nuevo panorama para las TI, ofreciendo innumerables beneficios para lograr la interoperabilidad entre los procesos de negocios de una empresa.

Oracle BPEL Process Manager es una herramienta que permite desarrollar, administrar y orquestar los sistemas empresariales, ya que está compuesta de múltiples tecnologías que hacen posible la interacción con los servicios web. Esto hace que la interacción con los procesos de negocios sea más fácil y flexible.

La orquestación de servicios web permite definir la lógica encargada de automatizar la interacción entre varios servicios y crear los procesos de negocio de una empresa.

Las tecnologías BPEL permiten reutilizar las aplicaciones existentes y también se pueden integrar sin problemas a los procesos de negocios, mejorando la escalabilidad y evitando los cuellos de botella en la ejecución de procesos. Esto mejora efectivamente el flujo de información en una organización.

## ABSTRACT


Service Oriented Architecture (SOA) is a new concept for IT, offering a number of benefits to reach the interoperability between the processes of businesses of a company.

Oracle BPEL Process Manager is a tool that permits to develop, administer and orchestrate the companies systems, because they are made of multiple technologies that make the interaction of the web services possible. This makes the interaction of the business process easier and flexible.

The orchestration of web services permits to define the logic in charge of automatizing the interaction between various services and to create the business processes of a company.

The BPEL technology permits to reuse existent applications and can also integrate the business process, improving the scalability and avoiding bottle neck in the process. This improves effectively the flow of information within an organization.



  
Translated by:  
Lic. Lourdes Crespo

# CAPITULO 1

## CONCEPTOS BASICOS

En este capítulo se determinan los conceptos básicos de una arquitectura orientada a servicios, seguido de la descripción de las características y arquitectura de *Oracle BPEL Process Manager*. Por último, se detallan los estándares, sistemas de integración y componentes de los servicios web.

### 1.1 Arquitectura Orientada a Servicios (SOA)

#### 1.1.1 Introducción

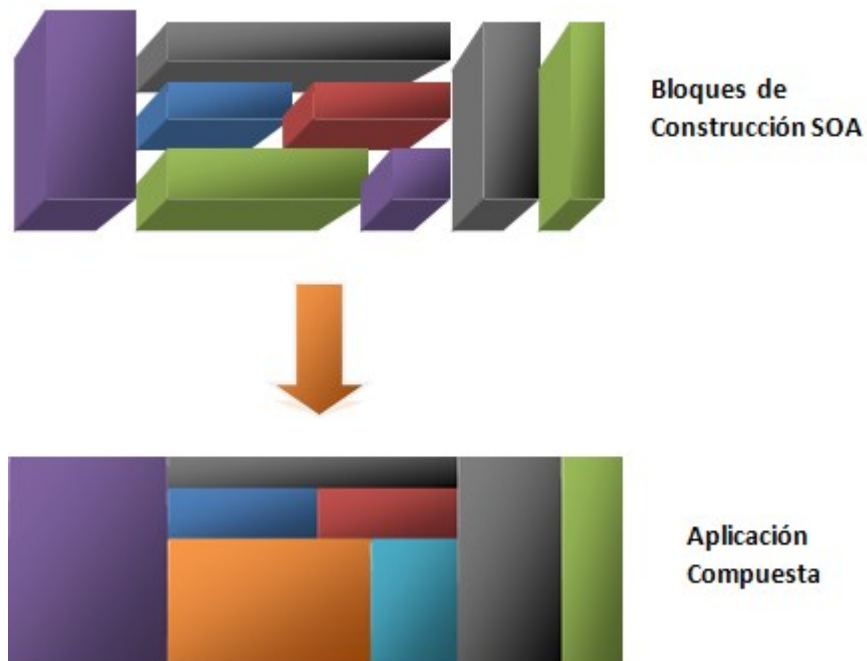


Figura 1.1: Arquitectura Orientada a Servicios. (Beecher, 2011)

En los últimos años, la evolución de las empresas mercantiles junto con las necesidades de los consumidores, están presionando a las TI para que ofrezcan mayor flexibilidad y rapidez en los procesos de negocio. En la actualidad, cada organización se enfrenta a las predicciones de cambios en un entorno global de mercados.



En respuesta a estos retos, las compañías líderes están adoptando arquitecturas orientadas a servicios (SOA), para cumplir con la oferta y demanda que se presentan continuamente, superando así la complejidad de sus aplicaciones y entornos de TI.

SOA proporciona una arquitectura empresarial que soporta la construcción de aplicaciones conectadas a la empresa, dando solución a los problemas empresariales. Facilita el desarrollo de sistemas empresariales, creando una infraestructura verdaderamente flexible y adaptable de TI, como por ejemplo un servicio web del módulo comercial que puede ser fácilmente integrado y reutilizado en varias aplicaciones. Es decir, se pueden mover las piezas y luego reconfigurarlas, convirtiendo los sistemas de TI en el equivalente a los bloques de Lego. **(Beecher, 2011)**

### **1.1.2 Capas y Componentes SOA**

Capas de una SOA

Las capas de una SOA según (Arsanjani, Borges y Holley 2004) son:

Capa 1

Sistemas operacionales: Esta capa contiene sistemas o aplicaciones existentes, como por ejemplo aplicaciones ERP, CRM o aplicaciones que utilizan las entidades públicas, sistemas heredados, implementaciones de sistemas orientados a objetos y aplicaciones de inteligencia de negocios. SOA reutiliza e integra estos sistemas existentes utilizando técnicas de integración orientadas a servicios. **(Arsanjani, 2004)**

Capa 2

Componentes empresariales: Esta capa se encarga de la funcionalidad y mantenimiento del servicio que se expone. Utiliza el diseño y la tecnología basada en el desarrollo de componentes. **(Arsanjani, 2004)**

### Capa 3

Servicios: En esta capa los servicios de negocio pueden ser expuestos o pueden ser enlazados estáticamente y después invocados en servicios compuestos. Permite seleccionar componentes empresariales de la capa anterior, componentes de unidades de negocio, y en algunos casos componentes específicos del proyecto, para externalizar un subconjunto de sus interfaces en forma de descripción de servicios, los cuales pueden ser simples o compuestos. **(Arsanjani, 2004)**

### Capa 4

Composición de procesos de negocio: En esta capa se define la composición de los procesos de negocio basados en los servicios de la capa anterior. Los servicios son implementados a través de la orquestación y coreografía para ejecutar los procesos de negocios. En la parte del diseño, se pueden recurrir a herramientas visuales de composición de flujos. **(Arsanjani, 2004)**

### Capa 5

Presentación o acceso: Esta capa normalmente se consideraba fuera del ámbito de SOA, pero debido a nuevos estándares como *Web Services for Remote Portlets* están comenzando a ser relevantes los servicios en la capa de presentación. Es preciso resaltar que SOA desmonta la interfaz de usuario de los componentes. **(Arsanjani, 2004)**

### Capa 6

Integración de servicios (ESB): Esta capa facilita la integración de servicios a través de la introducción de un conjunto tecnologías y capacidades integras, como por ejemplo el enrutamiento profundo y otros mecanismos de transformación, que a su vez se definen como Enterprise Service Bus (ESB). **(Arsanjani, 2004)**

## Capa 7

Calidad del servicio: Esta capa se encarga de la seguridad, ejecución y disponibilidad de los servicios, que a su vez, servirán para monitorizar, administrar y mantener propiedades particulares de los servicios. Es un proceso de *background* a través de mecanismos de solicitud-respuesta y herramientas que monitorizan el estado de las aplicaciones SOA incluyendo las implementaciones de *WS-Management (Web Services for Management)* y otros protocolos y estándares relevantes que implementan la calidad de servicio para una SOA. (Arsanjani, 2004)

En la siguiente figura se pueden ver las capas y componentes de una SOA según (Arsanjani, Borges y Holley 2004).

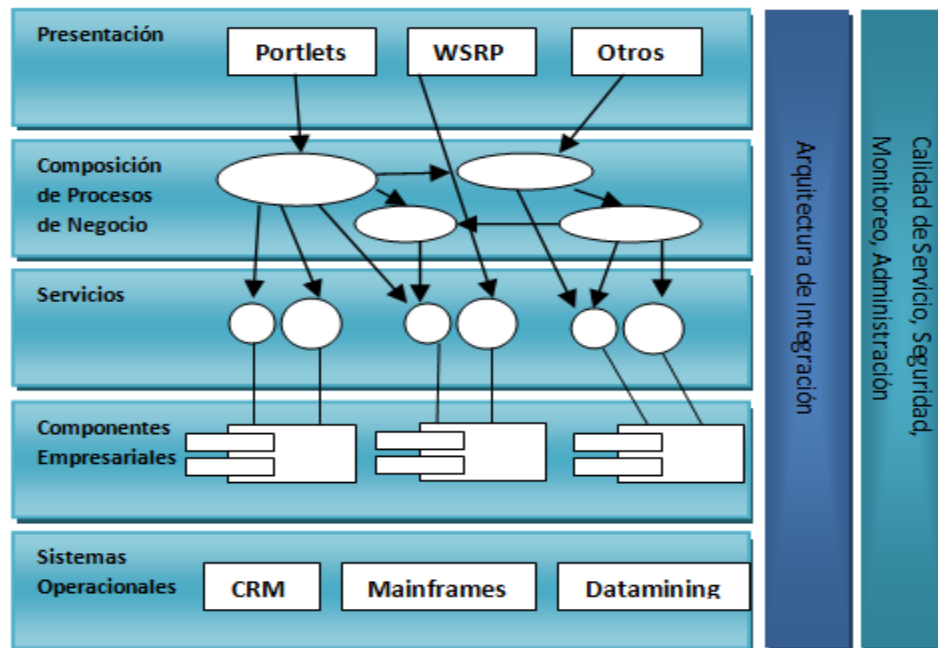


Figura 1.2 Capas de una SOA. (Arsanjani, 2004)

Componentes de una Arquitectura Orientada a Servicios (SOA). (Chao M Beck, 2011)

La implementación de una arquitectura SOA está conformada de los siguientes componentes tecnológicos:

- Enterprise Service Bus (ESB): Es un software que se encarga de facilitar la comunicación entre los servicios.
- Herramienta de desarrollo: Es un entorno integrado que se especializa en el diseño y construcción de la orquestación de los servicios.
- Administrador de Servicios: Es el encargado de mantener actualizado los servicios registrados, para ser consumidos por las diferentes aplicaciones de negocios.
- Servicios de Información: Son las funciones que gestionan los datos y contenidos de distintas fuentes de forma unificada.
- Servicios de Acceso: Son funciones que facilitan las interacciones entre las diversas aplicaciones del negocio.

### 1.1.3 Arquitectura de Servicios web

La arquitectura de servicios web se caracteriza por las interacciones de tres elementos: proveedor de servicios, registro del servicio y consumidor del servicio. Estas interacciones interpretan las operaciones de publicación, exploración y enlace. **(Dezhgosh, 2005)**

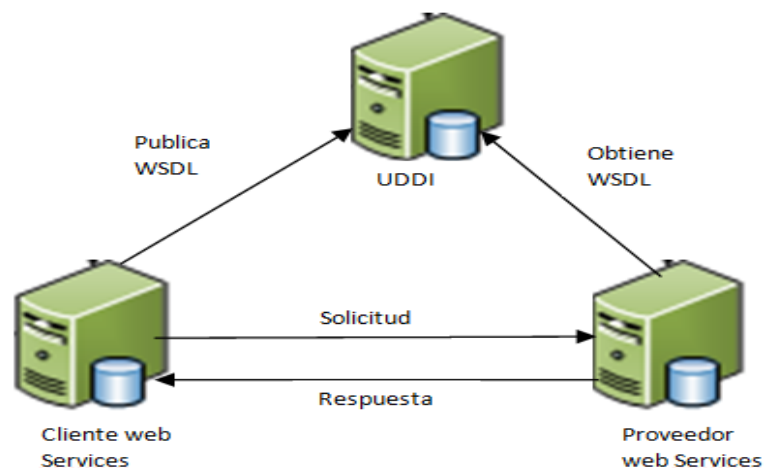


Figura 1.3 Arquitectura de Servicios Web. **(Dezhgosh, 2005)**

En una arquitectura de servicios web sobresalen los siguientes roles: **(Dezhgosh, 2005)**

- El Proveedor del servicio Web: Este rol integra su funcionalidad y la hace disponible.
- El Solicitante del servicio Web: Este rol consume los servicios que brinda el proveedor mediante el envío de requerimientos XML.
- El Directorio de servicios Web: Es la ubicación donde los proveedores ostentan sus servicios y donde los clientes los buscan para utilizarlos.

Los proveedores utilizan WSDL para especificar las interfaces externas de los servicios Web dados. El WSDL es utilizado para expresar:

- La localización de los *web services*.
- Las descripciones e invocaciones de los procesos que realizan los *web services*.
- Las funciones de negocio que procesa cada *web services*.

Estas descripciones son generalizadas en los registros de los *web services*, para que los clientes logren acceder a ellas y puedan utilizarlas.

Los proveedores y clientes se comunican por medio del protocolo de mensajes SOAP, que está escrito en XML. Este protocolo es consumido por los clientes para enviar requerimientos y por los proveedores para enviar las respuestas a estos. Este protocolo se utiliza para intercambiar mensajes y efectuar llamadas a procesos remotos, en un ambiente distribuido de forma transparente y sencilla. Hay dos tipos definidos de mensajes: Solicitud (utilizan los clientes) y Respuesta (utilizan los proveedores).

Para implementar los registros de servicios web opcionales, se lo hace mediante la utilización del protocolo UDDI, el cual es un grupo de protocolos que tienen como objetivo, proveer un directorio de servicios que pueden ser consultados tanto por clientes como por aplicaciones. Para descubrir los servicios web y obtener sus respectivas descripciones, se debe usar WSDL para invocarlos. **(Dezhgosh, 2005)**

Para lograr los objetivos de negocio mediante la interacción de varios servicios web, los desarrolladores utilizan BPEL, ya que es un lenguaje de orquestación de servicios web fundamentado en XML, definiendo así la forma en que van a cooperar los servicios web. Estos servicios web pueden ser definidos de manera independiente o no, para realizar un proceso. **(Madani, 2010)**

En definitiva, como BPEL está basado en XML, la programación puede realizarse de manera normal y de la misma forma en que se codifica un programa en XML, pero en el uso habitual, BPEL se programa de forma gráfica, para que los procesos de negocio implementados puedan ser definidos y analizados por los responsables del negocio.

#### **1.1.4 Ciclo de Vida SOA**

El ciclo de vida de una SOA se compone de cuatro fases, las cuales son respaldadas por un grupo de procesos de gobierno, que permiten asegurar que los cambios ocurran en un ambiente controlado y con la autoridad apropiada, como fue establecido en el diseño del negocio. Las modificaciones pueden realizarse debido a presiones de negocios externas o también cuando existen mejoras internas en el proceso en funcionamiento. **(Chao M Beck, 2011)**



Figura 1.4 Ciclo de Vida de SOA

Las fases del ciclo de vida de una SOA son: **(Chao M Beck, 2011)**

1. Modelado (*Model*): Esta fase recolecta y evalúa las necesidades del negocio, definiendo los procesos para después diseñar los servicios que darán soporte a estos procesos.
2. Ensamblado (*Assemble*): Durante esta fase se crean los servicios con los recursos que existen, como puede ser mediante ERP, CRM u otras herramientas que se manejen dentro del negocio. Por lo tanto, estos servicios se fusionan para construir los procesos de negocios.
3. Implementado (*Deploy*): En este ciclo se configura el entorno de ejecución para que los procesos de negocio alcancen los niveles de servicio requeridos, y puedan ser organizados dentro de un ambiente seguro y escalable. Es decir, si existen cambios en el negocio, se puede optimizar este ambiente de servicios, para ejecutar procesos críticos y obtener la flexibilidad de hacer actualizaciones dinámicas.
4. Gestionado (*Manage*): Esta fase se relaciona con el mantenimiento de los servicios disponibles, el establecimiento y los tiempos de respuesta. También, se basa en el monitoreo en tiempo real de los indicadores de rendimiento, obteniendo la información necesaria para prevenir, bloquear, diagnosticar y resolver problemas en los procesos de negocio.

## **1.2 Oracle BPEL Process Manager**

### **1.2.1 Introducción**

BPEL, es utilizado para especificar el uso de los distintos procesos de negocio basados en web services. Es un lenguaje de orquestación de web services basado en XML, el cual especifica la forma en que van a cooperar los servicios web entre las diferentes soluciones de la empresa. BPEL reduce la complejidad y el costo de los procesos de negocio mediante la integración de una serie de servicios dentro de un sencillo flujo de proceso. (Madani, 2010)

*Oracle BPEL Process Manager* es parte de la familia de *Oracle Fusion Middleware*, el cual ayuda a desarrollar y a ofrecer una Arquitectura Orientada a Servicios o SOA de una forma rápida.

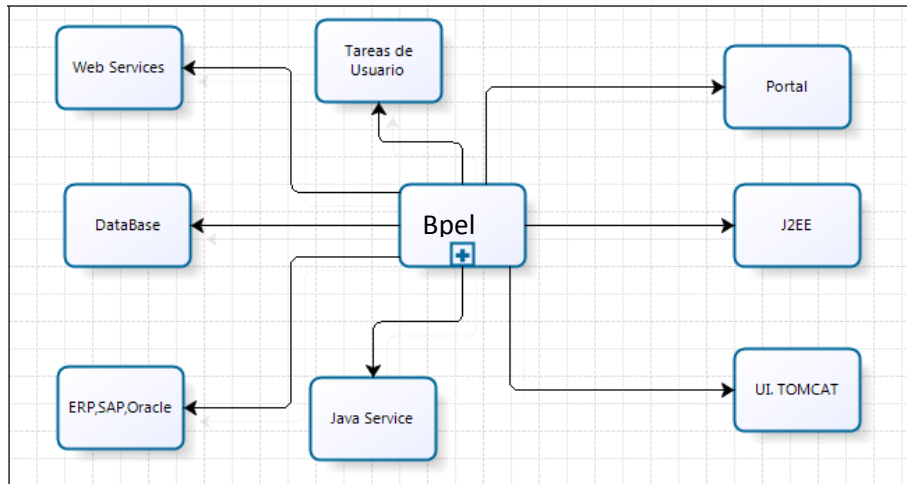


Figura 1.5 Oracle BPEL Process Manager.

*Oracle BPEL Process Manager* provee un estándar para acoplar a un conjunto de servicios diferenciados en un flujo de procesos de fin a fin, reduciendo fundamentalmente el costo y la complejidad de los procesos principales de integración, ya que permite organizar los servicios sincrónicos y asincrónicos en los flujos de proceso BPEL. (Oracle, 2010)

Los procesos BPEL se integran con los servicios externos conocidos como (*PartnerLinks*). También, integran adaptadores de tecnología y servicios, tales como las tareas humanas (*HumanTask*), transformaciones, notificaciones y reglas de negocio dentro de los procesos.

### 1.2.2 Características

El procesamiento en paralelo, los resultados en tiempo de espera y las notificaciones son algunas de las características importantes de *Oracle BPEL Process Manager*, ya que emplea estándares BPEL para el diseño, implementación, monitoreo y administración de procesos.

Los estándares de servicios web, tales como WSDL, SOAP y XML que es característico de BPEL, pueden administrar las excepciones y los errores de manipulación al mismo tiempo, tanto en el momento de diseñar una aplicación o durante el tiempo de ejecución. Permite mantener la fiabilidad



y la escalabilidad de los procesos, ya que se guarda de forma automática el estado de flujo de ejecución en la base de datos. (Oracle, 2004).

*Oracle BPEL Process Manager* se puede instalar en varios sistemas operativos y se integra fácilmente a diferentes tipos de servidores de aplicación como son: *JBoss*, *BEA WebLogic*, *Oracle Application Server*, etc. Contiene un diseñador gráfico de procesos de fácil manejo para el usuario, poniéndolo disponible como conexión a Oracle BPEL JDeveloper. También, puede ser integrado con otras aplicaciones como Oracle APEX Interconexión de Integración, Oracle *Workflow*, Oracle *Business Activity Monitoring*, y la integración de Empresa a Empresa en Oracle *Application Server* a través de adaptadores JCA y protocolos nativos. (Oracle, 2004)

### 1.2.3 Arquitectura

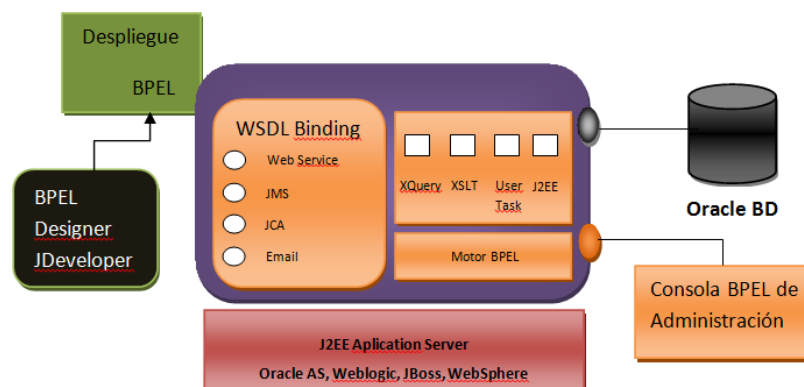


Figura 1.6 Arquitectura de Oracle BPEL Process Manager. (Kennedy, 2007)

Oracle BPEL Process Manager es una aplicación J2EE y se ejecuta en un contenedor J2EE como por ejemplo JBoss y OC4J. Para que los procesos BPEL se inicien, los servicios web deben hacerlo mediante llamadas a través de HTTP o por ciertos procesos auto-invocados por la recepción de un sistema de archivos. (Kennedy, 2007)

Un proceso BPEL es invocado, llamado o ejecutado en una nueva instancia. En la mayoría de procesos BPEL, se tienen muchas instancias diferentes, es decir, algunos pueden estar en funcionamiento paralelo o ejecutándose al mismo tiempo, para lo cual se recibirán muchas órdenes o solicitudes, como por ejemplo de reservas para habitaciones, servicios de consultoría, etc, todas al mismo tiempo. (Kennedy, 2007)

La ejecución de una instancia de un proceso BPEL implica: la llamada a los servicios web, la espera de las respuestas, el resultado del rendimiento de un tipo de procesamiento lógico y la invocación de más servicios para calcular el resultado final. Los servicios invocados durante la ejecución de un proceso pueden ser las tareas del usuario o flujos de trabajo, las cuales son monitoreadas y registradas en cada etapa de su ejecución.

Toda la información sobre las instancias de procesos individuales, tales como, su situación actual y los intercambios de mensajes con los servicios externos, están disponibles a través de la consola de BPEL, ya que este es un navegador basado en aplicaciones web.

Los ensayos de los distintos casos de procesos se pueden iniciar a través de la consola de BPEL e incluso una prueba de carga de estrés se puede ejecutar. Esta consola proporciona información estadística de todas las instancias ejecutadas, como por ejemplo puede mostrar la instancia de un proceso de depuración y los valores de sus variables ejecutadas en cada etapa del proceso. (Kennedy, 2007)

*Oracle BPEL Process Manager* almacena la información sobre la ejecución de instancias en una base que soporte JDBC, ofreciendo así la capacidad de "deshidratación" que hace que el estado de flujos de ejecución prolongados se guarde automáticamente. Es decir, si la ejecución de un proceso en una instancia dura más de lo esperado, el motor de BPEL hace que se serialicen los estados de esa instancia. Esto ayuda a minimizar el uso de recursos del sistema, ya que al momento de ejecutarse un proceso prolongado, BPEL calcula el umbral para determinar el tiempo que esperara y tomar la decisión de deshidratar o no la instancia. (Kennedy, 2007)

Las ventajas de esta deshidratación en la base de datos son: la escalabilidad y fiabilidad. (Kennedy, 2007)

- **Fiabilidad:** Es cuando en una instancia el motor del servidor BPEL se bloquea, el estado del proceso no se pierde, sino, se encuentra disponible en la base de datos y pueden, en gran medida, restablecerlo desde ese punto.
- **Escalabilidad:** Es cuando queremos implementar múltiples instancias de *BPEL Process Manager* en varios servidores de aplicaciones y tener un equilibrio de carga entre ellos. Es decir, que un proceso puede mantenerse iniciado en una instancia del servidor de BPEL, el cual decidirá si deshidrata la instancia y a la vez, se rehidrata en una nueva instancia. Para que se rehidrate la instancia, el motor BPEL toma el último estado almacenado en la base de datos y completa la ejecución.

#### **1.2.4 Servicios de Integración**

##### **Integración con Oracle generalizado**

Los servicios de integración se procesan a través de Oracle *Application Server*, ya que permite el acceso rápido a los datos heterogéneos y ofrece la capacidad única de procesar el contenido no-relacional y semi-estructurado que se encuentran internamente en la mayoría de las empresas. Los adaptadores de estos servicios, eliminan la complejidad de codificación de una interfaz de bajo nivel, reduciendo así costos, tiempo y riesgo. (Oracle, 2009)

Generalmente, Oracle *Application Server* transforma y crea servicios compuestos de diferentes fuentes y proporciona accesibilidad total a Oracle BPEL *Process*. La flexibilidad de Oracle *Application Server* ofrece una amplia gama de opciones de conectividad para varias aplicaciones de Oracle, como Oracle *Financials*, Siebel CRM, etc. (Oracle, 2009)

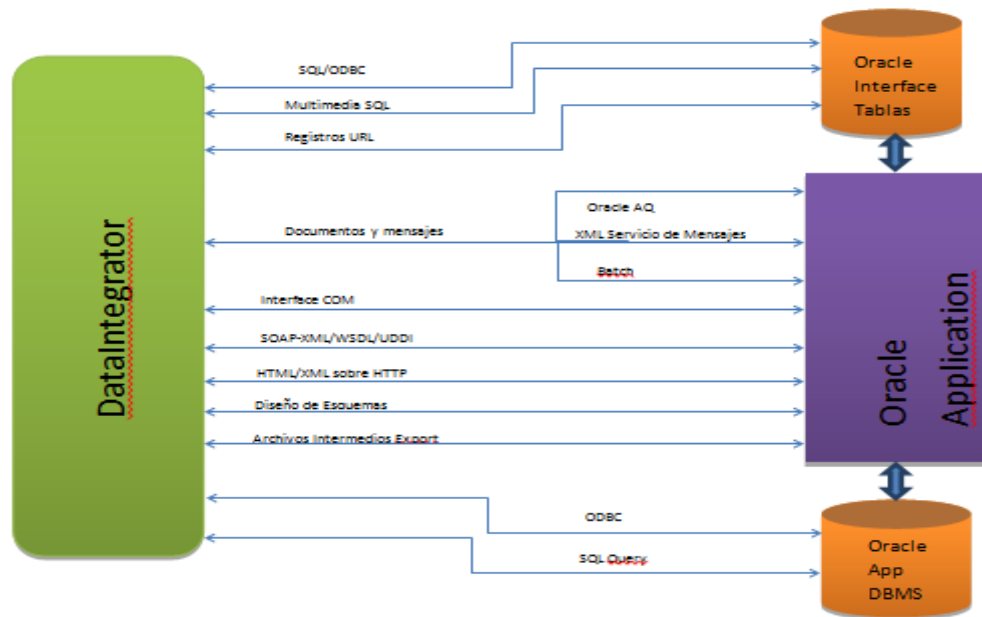


Figura 1.7 Integración de Oracle en los procesos de negocio. (Oracle, 2009)

Oracle *Data Integrator* y Oracle SOA Suite permite a las empresas realizar cualquier tipo de transferencia de datos y las transformaciones en tiempo real, ya que pueden fácilmente gestionar las iniciativas relacionadas con la inteligencia de negocios (BI), *Data Warehouse*, gestión de datos maestros (MDM), Oracle BAM, la migración y consolidación de aplicaciones y SOA. (Oracle, 2009)

Para transformar la carga de un documento XML, un proceso BPEL o un mediador de componente de servicio, pasa la carga de XML de más de 10 MB por Oracle *Data Integrator* para transformarlo. El resultado será la creación de otro archivo XML que se almacena en una base de datos. También, orquesta el nivel de datos y la inserción en un área de ensayo, transformando después los datos agregados y las tablas de carga. Esto permite a Oracle *Business Intelligence* y a las aplicaciones de almacenamiento, informarse sobre los datos activos. (Oracle, 2009)

## 1.3 Web Service

### 1.3.1 Introducción

Un *web services* es un conjunto de funciones empaquetadas dentro de una aplicación disponible en otros sistemas a través de la red. Las redes pueden ser intranets corporativas o de internet. (Oracle, 2009). También, se puede decir que es un sistema diseñado para soportar la interoperabilidad entre los sistemas de interacción en la red, ya que cuentan con una interfaz descrita en un formato WSDL estructurado mediante el lenguaje XML. Otros sistemas interactúan con los servicios web en forma prescrita por su descripción, ya que utilizan SOAP, para transmitir sus mensajes a través de HTTP con una serialización XML en conjunto con otros estándares web familiarizados.

### 1.3.2 Estándares Relacionados

Los servicios web se basan en ciertos estándares de comunicación, como por ejemplo XML para la representación de datos, SOAP para el intercambio de datos y el lenguaje WSDL para describir las funcionalidades de un servicio web. (W3C, 2004)

Existen varias formas de visualizar estos estándares, así como hay distintas maneras de construir y utilizar los servicios Web.



Figura 1.8 Diagrama de pila de Estándares Relacionados. (W3C, 2004)

El esquema consta de tres partes para mostrar cómo los servicios Web se relacionan entre sí, por ejemplo: WSDL describe el formato de los mensajes SOAP y UDDI se utiliza como un servicio de localización de las descripciones WSDL. El lenguaje XML ayuda a borrar la distinción entre "carga" de datos y "protocolo", lo que facilita la interconexión entre protocolos de comunicación diferentes. Esto es necesario en muchas empresas para implementar infraestructuras de TI que se basan en la fortaleza industrial. (W3C, 2004)

Los servicios se invocan y proporcionan los resultados mediante mensajes, por lo que deben ser intercambiados a través de algún medio de comunicación como: HTTP (el protocolo dominante de la "Web"), otros protocolos de Internet como SMTP, FTP y APIs de interfaz genérica como JMS. (W3C, 2004)

### **1.3.3 Integración en los Procesos de Negocio**

El orden de ejecución potencial de las operaciones de un conjunto de servicios web se especifica en un proceso de negocios. Esto a la vez permite describir las transacciones de larga duración entre los servicios web, aumentando la consistencia y fiabilidad para las aplicaciones. Por ende, se rompe con la confusión de los servicios web, ya que en ocasiones representan restricciones sobre cómo las operaciones de un conjunto de servicios web y su conjunto de comportamientos pueden ser utilizados.

BPEL4WS o BPEL permite especificar los procesos de negocio y saber cómo se relacionan con los servicios web. Un proceso de negocios hace uso de los servicios web para lograr su objetivo, determinando también como los servicios web proporcionan un proceso de negocio para lograrlo. Mediante BPEL, los procesos de negocios desarrollados son totalmente ejecutables y portátiles entre entornos, ya que un proceso de negocio BPEL interactúa con los servicios web de sus socios. Estos servicios web se ejecutan en base a BPEL, soportando la especificación de protocolos de negocios entre los socios y los complejos procesos empresariales. (W3C, 2004)

### 1.3.4 Composición de servicios web

Definición de las características de los elementos que intervienen en un servicio WEB.

#### WSDL

WSDL (*Web Service Description Language*) mediante XML especifica una gramática que permite describir los servicios web como un conjunto de nodos de comunicación, ya que tienen la capacidad de intercambiar mensajes entre servicios. Es decir, un documento WSDL define los servicios como una colección de nodos de red o puertos, debido a que los protocolos de comunicación hacen posible la estandarización de los mensajes en la web. (Oracle, 2009)

Un archivo WSDL debe utilizar los siguientes elementos para definir un servicio WEB: (Louridas, 2008)

- Servicio: Es un conjunto de nodos comúnmente relacionados.
- Puerto: Es la combinación de una vinculación y una dirección de red referentes a un nodo.
- Tipos: Son metadato que se utilizan en el sistema como por ejemplo XSD.
- Mensaje: Son los datos definidos de una manera abstracta que son transmitidos.
- Tipo de Puerto: Son las operaciones abstractas que soportan los puertos o nodos.
- Operación: Representación abstracta de las operaciones soportadas por un servicio.
- Vinculación: Es la especificación del formato de datos de un tipo de puerto con su respectivo protocolo.

La siguiente figura muestra cómo se relacionan los elementos de WSDL:

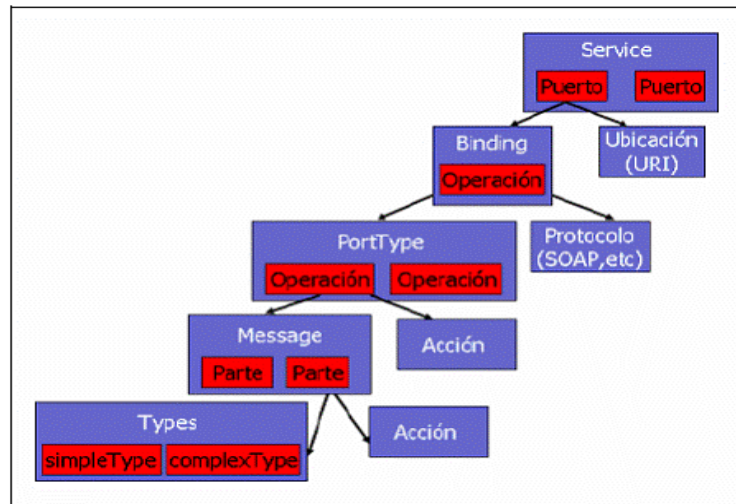


Figura 1.9 Relación entre los elementos de WSDL. (Louridas, 2008)

## SOAP

SOAP (*Simple Object Access Protocol*), es un protocolo especializado en el intercambio de información en sistemas distribuidos, el cual utiliza un conjunto de estándares para empaquetar mensajes, ya que utiliza XML y soporta varios protocolos en una estructura de mensajes SOAP. Básicamente se basa en la extensibilidad y la simplicidad para definir la estructura de los mensajes SOAP. (W3C, 2004)

Dentro de las ventajas de SOAP es posible distinguir las siguientes: (Louridas, 2008)

- SOAP no asocia con protocolos, ya que es un documento XML.
- SOAP tampoco está ligado a ningún lenguaje.
- SOAP no está relacionado con ninguna infraestructura de objetos distribuidos.
- SOAP utiliza estándares existentes en cualquier entidad de negocio, extendiéndolos concurrentemente, como por ejemplo XML.
- SOAP también posibilita la interoperabilidad entre varios entornos.



## UDDI

UDDI (*Universal Description, Discovery, and Integration*) facilita la búsqueda de servicios dentro de SOA, proporcionando un método estandarizado para la publicación y el descubrimiento de información dentro de los Servicios Web. Existe una gran parte del comercio electrónico que está basado en los Servicios Web, por ende las empresas deben encontrar estos servicios por sus propios medios de búsqueda, lo cual es una tarea muy compleja y difícil. (W3C, 2004)

UDDI es un registro conceptual único, porque se encuentra distribuido en los servicios web existentes. Para registrar ciertos tipos de información, los proveedores desarrollan un Servicio Web para incorporarlo a la UDDI de manera que pueda ser localizado fácilmente, como por ejemplo: (Louridas, 2008)

- Páginas amarillas: Facilita la búsqueda, ya que utiliza una clasificación específica para los servicios web.
- Páginas Verdes: Describe el funcionamiento y comportamiento de los servicios web, ya que contienen la información técnica de los mismos.
- Páginas blancas: Identificar y contacta a la empresa que ofrece el servicio, ya que contiene la información básica de estas.

UDDI es un motor de búsqueda específico para los procesos de negocios según los analistas. Lo que posibilita la realización de tareas para los desarrolladores que desean publicar un servicio en una API. También, hay portales Web que interactúan con los desarrolladores y quienes buscan un servicio específico, donde se facilitan las búsquedas para registros especiales, teniendo un carácter más general en varios de estos registros.

La siguiente figura muestra esta relación entre quien busca y quien ofrece.

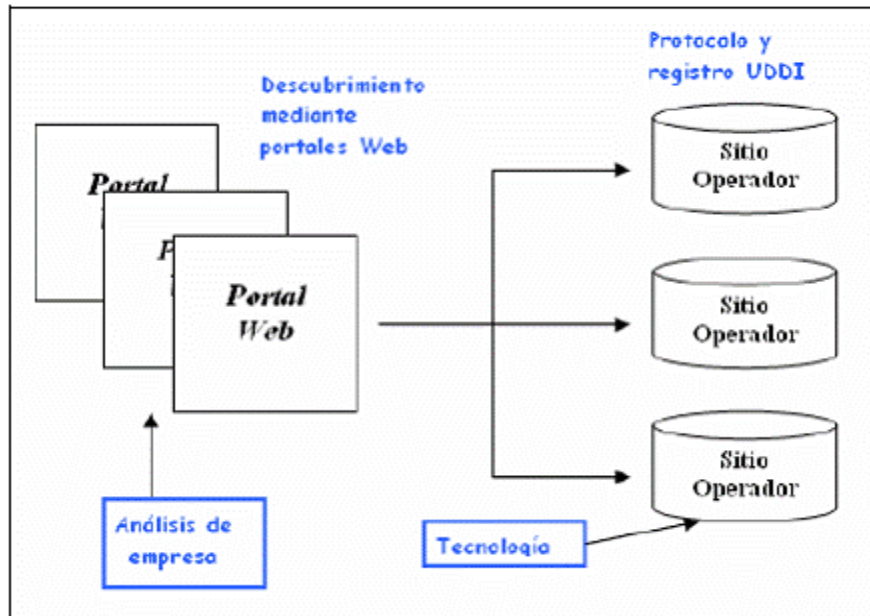


Figura 1.10 Relación entre tecnología y negocios en UDDI. (Louridas, 2008)

## HTTP

HTTP (*Hypertext Transfer Protocol*) es utilizado por los Servicios Web como mecanismo de comunicación, ya que es un protocolo estándar para la transferencia de documentos en Internet. (W3C, 2004)

## XML

XML (*eXtensible Markup Language*) es un lenguaje diseñado para representar y transferir datos estructurados. (W3C, 2004)

Como conclusión de este capítulo, se puede decir que los *web services* son las piezas claves para que los sistemas interactúen a través de la red, ya que son formados mediante WSDL y con la ayuda de otras tecnologías, hacen que los procesos de negocios hagan uso de ellos para lograr sus objetivos.

## CAPITULO 2

### INTEGRACION CON ORACLE BPEL PROCESS MANAGER Y WEB SERVICES

En este capítulo se describe lo que es la orquestación de servicios, para que sirven y cómo interactúan los procesos de negocios con las actividades básicas de BPEL. Además, se detalla la invocación de los servicios web, mediante los tipos de tecnologías y aplicaciones que se integran para determinar la lógica en una solicitud de servicios.

#### **2.1 Orquestación de Servicios**

##### **2.1.1 Introducción**

La Orquestación de servicios, es un proceso central que toma el control sobre todos los servicios conectados para un mismo objetivo. A través de un determinado nivel de mensajes, coordina la ejecución de diferentes operaciones en los servicios expuestos, incluyendo una lógica de negocios en el orden de ejecución de cada interacción. (Matjaz Juric, 2010)

Los servicios implicados no saben (y no deben saber) que están envueltos en una composición para ser parte de un proceso de negocios. Solo la orquestación centralizada se encarga de mapear los procesos de negocio por medio de un flujo funcional, que involucra la búsqueda de las herramientas compatibles con cualquier lenguaje. Es decir, se realiza un cruce entre herramientas de agentes externos y agentes internos de cualquier ente organizacional que las emplee. Esto hace que se genere una tabla de evaluación de las herramientas disponibles, para la orquestación de servicios web. (Matjaz Juric, 2010)

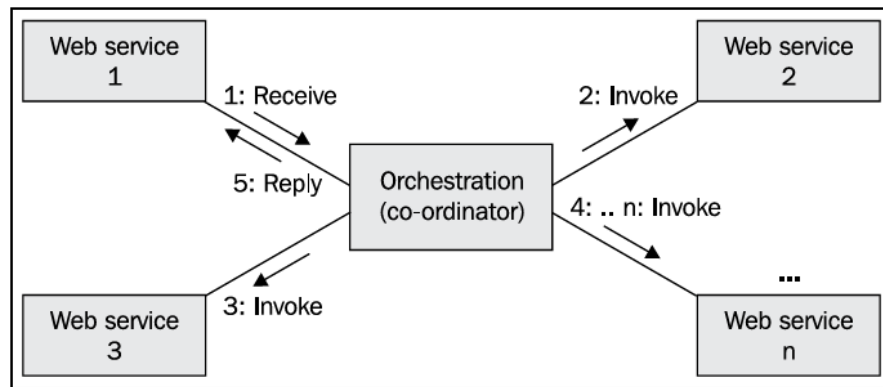


Figura 2.1 Orquestación de Servicios. (Matjaz Juric, 2010)

### 2.1.2 Estructuras de un Servicio BPEL

BPEL se divide en dos partes diferentes. La primera parte hace referencia a la parte externa de los procesos BPEL, el cual utiliza ciertos elementos para modelar la relación de comunicación par a par, como por ejemplo los *PartnerLink* o Socio de Enlace, que está asociado con el archivo WSDL. El *PartnerLink* es utilizado para describir las interacciones entre los procesos BPEL y el mundo exterior. La segunda parte se refiere a la parte interna de BPEL, que en este caso son las actividades que representan estructuras de control complejas, describiendo así a los procesos que serán ejecutados en un tiempo determinado. (Matjaz Juric, 2010)

En la siguiente figura se puede ver la estructura básica de un proceso BPEL, y como este se relación con ciertos componentes.

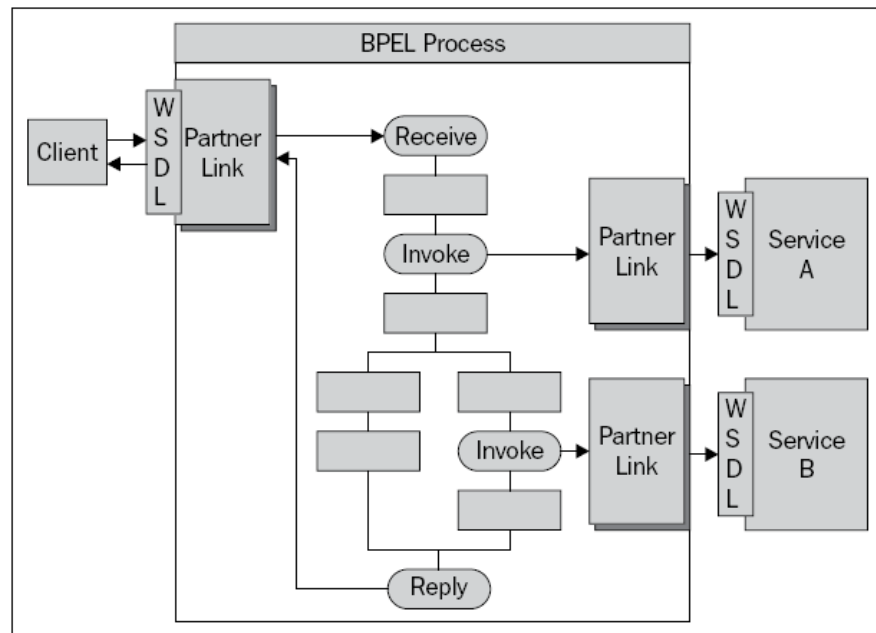


Figura 2.2 Estructura de un Servicio BPEL. (Matjaz Juric, 2010)

### 2.1.3 Elementos Básicos

#### PartnerLink

Básicamente, un *PartnerLink* es creado para definir un servicio externo, porque puede ser expuesto para interactuar con otros servicios compuestos, por ejemplo un *PartnerLink* de “atención al cliente” podría agrupar el servicio de consulta de saldos, compras y descuentos. Por tanto, describe los roles de un proceso o servicio y por ende los datos que se pueden manipular en ese rol. Un *PartnerLink* se define por su *PartnerLinkType*, el cual se caracteriza por su relación de interacción entre los servicios expuestos; definiendo los roles que pueden procesar los servicios mediante los *PortType*. Los *PortType* proveen a los servicios los mensajes de solicitud-respuesta dentro de una llamada entre dos servicios. (Matjaz Juric, 2010)

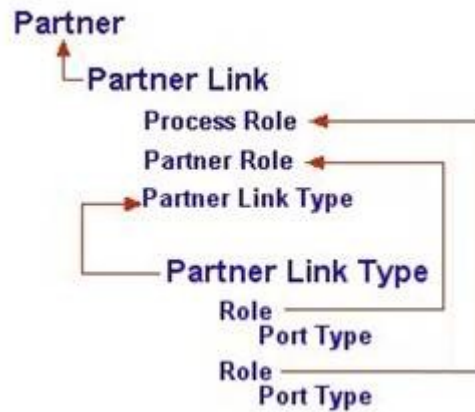


Figura 2.3 Relación Entre *Partners*, *Partner Links* y *Partner Link Types*

Los *PartnersLinks* están ligados al archivo WSDL del proceso en sí y a los *web services* mediante los tipos de enlaces de servicios.

Un proceso BPEL describe el flujo de interacciones entre los servicios y procesos. Cada interacción determina qué roles desenvuelven los procesos y servicios en cada flujo y que datos pueden ser manipulados en esos roles.

El uso de las definiciones de *Partners* en un proceso BPEL no hacen referencia a servicios web específicos, sino que un proceso BPEL es una definición reusable que puede ser desplegada de diferentes maneras, ya que las definiciones son abstractas. Por ende, la seguridad, el direccionamiento y otros componentes de los servicios web se toman en cuenta en el momento del despliegue. Todo Servicio Web involucrado en el proceso BPEL se modela como un *PartnerLink*, el cual permite especificar los servicios externos que el servicio BPEL va a interactuar. Cada *Partnerlink* se caracteriza por un *partnerLinkType*, aunque varios *Partnerlink* puede estar definido por un mismo *partnerLinkType*. En la siguiente imagen se visualiza la definición de un *Partnerlink*: (Matjaz Juric, 2010)

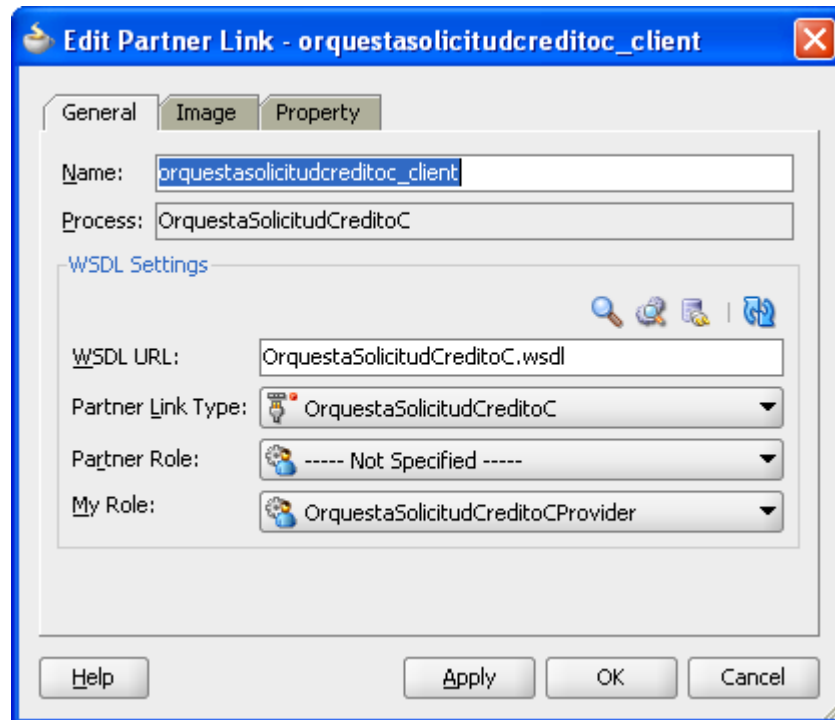


Figura 2.4 Diálogo PartnerLink

## Variables

Se emplean para el envío y recepción de mensajes por medio de los *partners* durante la ejecución del proceso BPEL. Estos tipos de variables persisten en largas interacciones, ya que son definidas en los tipos de mensajes WSDL.

En un proceso BPEL se puede enviar y recibir mensajes entre un par de *Partners* y guardar el estado de los mismos en variables. Para esto, incluye mensajes pasados desde y hacia procesos BPEL, mensajes intercambiados con servicios externos y variables locales usadas un flujo del proceso. Cada segmento de datos en BPEL es XML y el tipo de cada variable puede ser un tipo simple *Schema XML*, un elemento *Schema XML* o un tipo de mensaje WSDL. En la estructura BPEL las variables se definen de la siguiente manera: (Matjaz Juric, 2010)

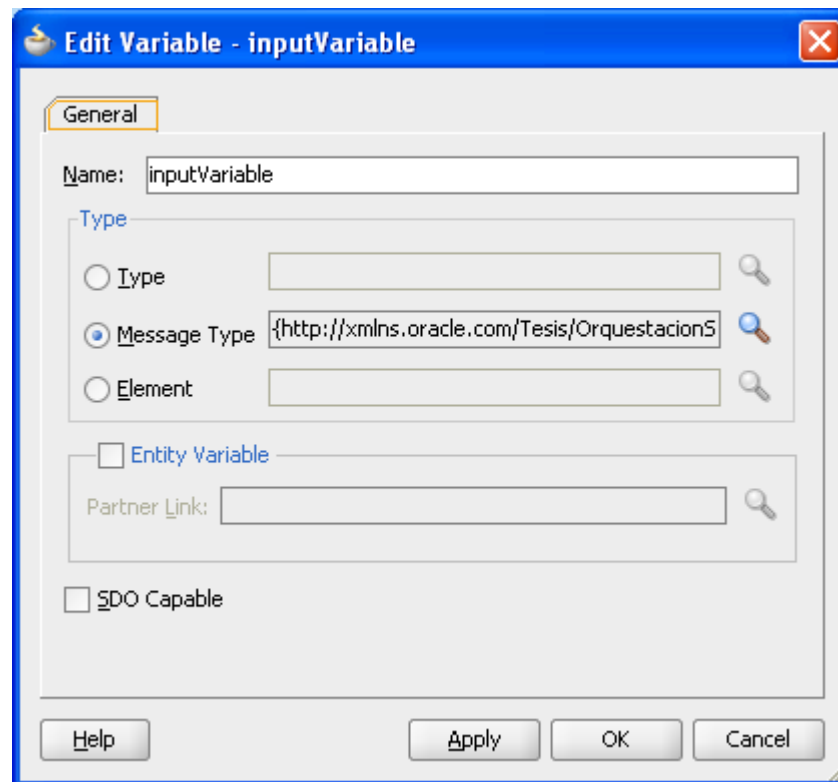


Figura 2.5 Diálogo Variable

### **Operaciones**

Son los tipos de servicios web que se requieren en los tipos de enlaces y son invocadas por medio de los mensajes. Estos tipos de operaciones pueden ser *receive* o *invoke*. (Matjaz Juric, 2010)

### **Port types**

Son los tipos de conexiones de servicios web necesarias en los enlaces con lo partners. (Matjaz Juric, 2010)

### **Sequence**

Es la que define toda la lógica en la orquestación de un servicio web. (Matjaz Juric, 2010)



## 2.1.4 Actividades Básicas

Las actividades permiten realizar distintas acciones dentro de un proceso, definiendo el proceso a modelar. Existen varios tipos de actividades, cada una de ellas con un conjunto de atributos que las caracterizan. (Matjaz Juric, 2010)

### Receive



Para hacer uso del proceso BPEL, esta actividad debe esperar por un mensaje que llegue de un cliente, por tal razón debe especificar el nombre del *partnerLink* que es el que tiene que esperar por el mensaje, el *portType*, y la operación que se pretende invocar. Además, para recibir el mensaje se puede especificar una variable, para tener una mejor referencia. En la siguiente figura se define la actividad *Receive*: (Matjaz Juric, 2010)

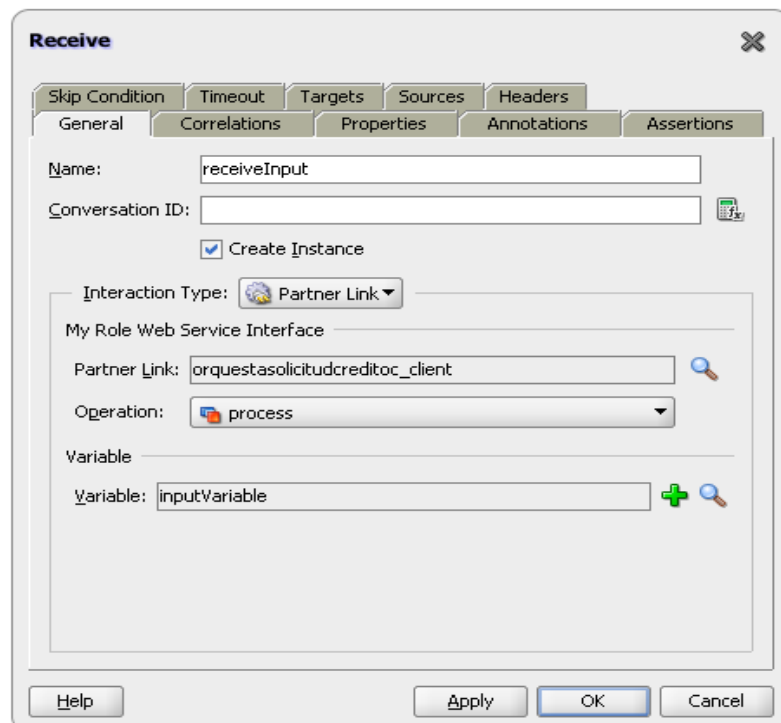


Figura 2.6 Diálogo Receive

Por lo general en un proceso *WS-BPEL* la actividad *<receive>* es la primera en ser activa, pero en una instancia de un mismo proceso no debe tener más de una actividad *receive* para un mismo *PartnerLink*, *PortType*, conjuntos de correlación y operaciones. (Matjaz Juric, 2010)

## Reply



A través de una actividad *reply* se envía un mensaje en respuesta a otro mensaje recibido. Para comunicarse con el proceso *WS-BPEL*, se debe combinar las actividades *receive* y *reply*, siempre y cuando la interacción se realice de manera asincrónica, ya que la actividad *reply*, puede especificar una variable que contenga el mensaje a enviarse como su respuesta. (Matjaz Juric, 2010)

Esta actividad solo tiene sentido si es precedida por una actividad *receive*. En la siguiente imagen se define la actividad *Reply*.

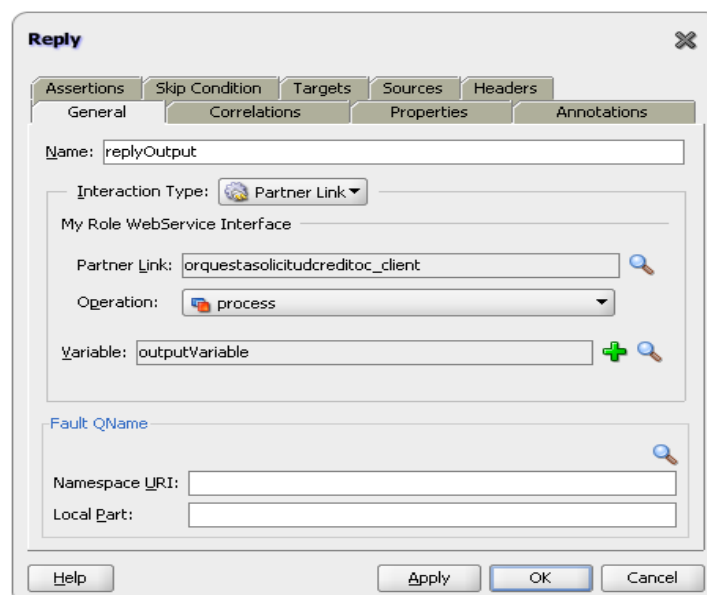


Figura 2.7 Diálogo Receive

## Invoke



Invoke1

Especifica la operación que el servicio debe realizar, invocando a un servicio asincrónicamente o de forma sincrónica (identificado por un *partnerlink*). La invocación asincrónica requiere solo la variable de entrada porque no espera ninguna respuesta, en cambio la invocación sincrónica, requiere una variable de entrada y otra de salida para recibir una respuesta casi inmediata por parte del *partner*, especificándose así uno o más conjuntos de correlación. (Matjaz Juric, 2010)

Si un servicio web falla en una invocación asincrónica, se producirá una falla en el proceso *WS-BPEL*, propagándose hacia la actividad que la contiene, ya que la actividad no tiene programado capturar la falla.

El manejador de compensación que se relaciona con las actividades asociadas, puede invocarse explícitamente o por medio del manejador en su entorno más cercano. La siguiente figura define la actividad *Invoke*.

The screenshot shows the 'Invoke' dialog box with the following configuration:

- Name:** Invoke6
- Conversation ID:** (empty)
- Detail Label:** (empty)
- Invoke as Detail
- Interaction Type:** Partner Link
- Partner Role Web Service Interface:**
  - Partner Link:** Oracle
  - Operation:** insert
- Variables:**
  - Input:** Invoke6\_insert\_InputVariable
  - Output:** (empty)

Figura 2.8 Diálogo Invoke

## Assign



Assign1

Esta actividad permite manipular datos, como por ejemplo, copiar el contenido de una variable a otra. La actividad de asignación, es el punto referente para la manipular datos en *WS-BPEL*, tomando *XPath* como base estándar, a través de expresiones, funciones y *queries*. También, para manipulaciones o transformaciones más complejas de datos es posible usar *XQuery*, *XSLT* o *Java*. (Matjaz Juric, 2010)

El origen y el destino de los datos a manipular, es proporcionado por el elemento *copy* de la actividad, permitiendo copiar de manera más específica el contenido del elemento fuente (variable, expresión, fragmento *XML* o *partner link*), desde la fuente (*from-spec*) al destino (*to-spec*). En la siguiente imagen se puede ver la actividad *Assign*.

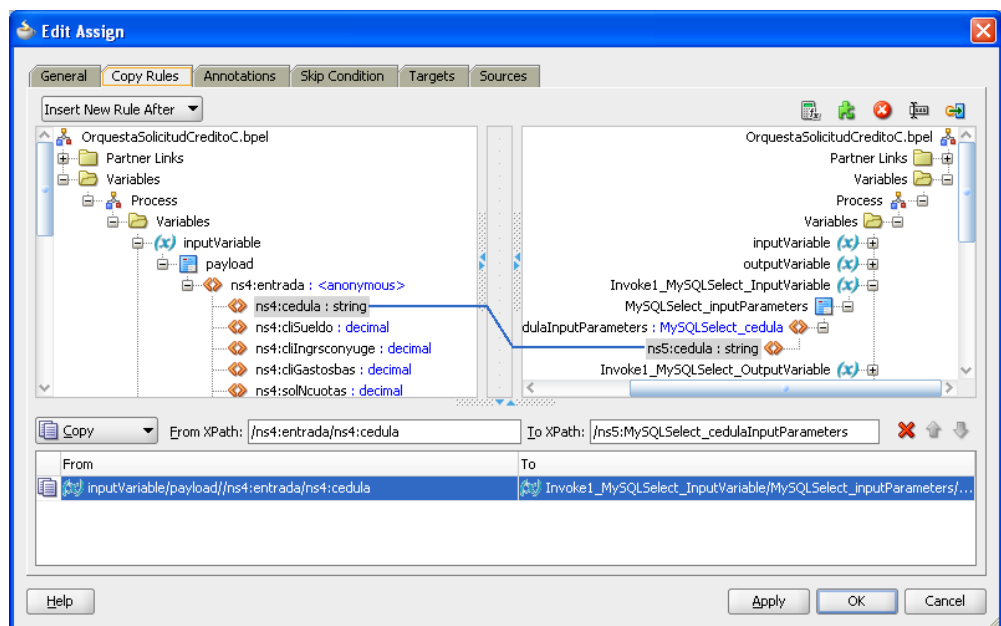


Figura 2.9 Diálogo Assign

## Throw



Throw1

Esta actividad se utiliza cuando existe una falla interna y el proceso WS-BPEL necesita indicarla explícitamente. Además, lleva la información adicional respecto al fallo ocurrido, por medio de una variable específica del atributo faultVariable. En la siguiente gráfica se define la actividad Throw. (Matjaz Juric, 2010)

Figura 2.10 Diálogo Throw

## Terminate



Terminate1

Se usa solo en procesos ejecutables, ya que su función es terminar la ejecución de un proceso por razones derivadas del manejo de eventos. (Matjaz Juric, 2010)

## Wait



Wait1

Esta actividad permite esperar un cierto periodo de tiempo que haya pasado. Mediante expresiones *XPath* se logra especificar las expresiones asociadas a los atributos *for* y *until*. En la siguiente figura se muestra la actividad *Wait*. (Matjaz Juric, 2010)

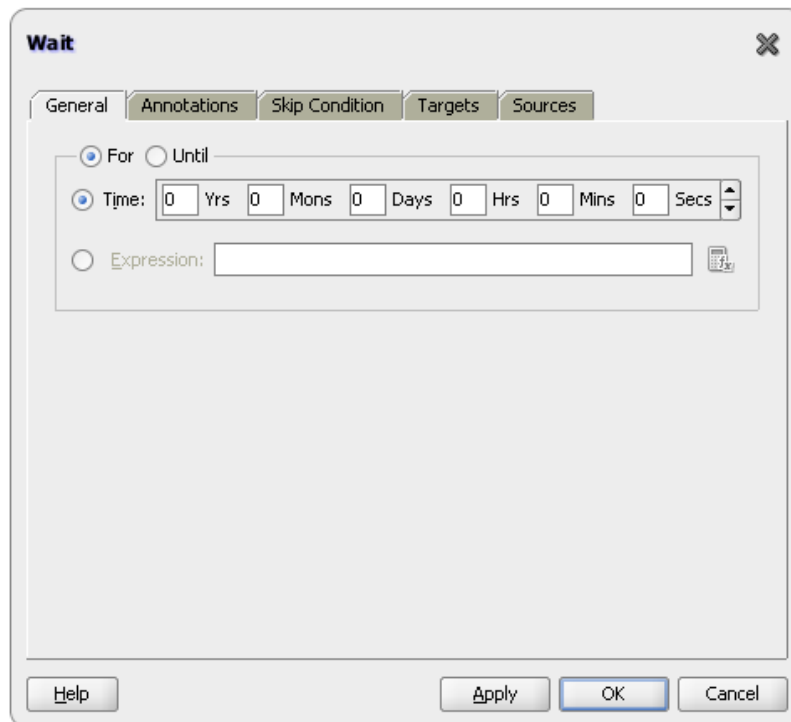


Figura 2.11 Diálogo Wait

## Empty



Empty1

Permite insertar una instrucción que no hace nada en los flujos de procesos y es utilizada para sincronizar actividades concurrentes. En el siguiente cuadro se ve la actividad *Empty*. (Matjaz Juric, 2010)

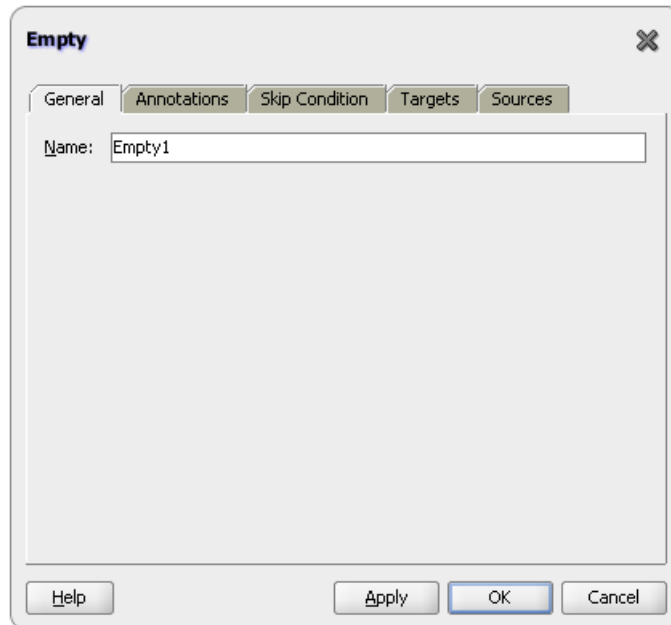


Figura 2.12 Diálogo Empty

### 2.1.5 Invocación de Servicios

En un proceso BPEL se debe mencionar los servicios de mensajes o llamadas de interacción que pueden tener las actividades `<receive>`, `<invoke>` y `<reply>`, estas a la vez pueden ser síncronas o asíncronas, que dependerán del uso del servicio invocado, ya sea por el consumidor o proveedor del servicio.

#### Invocación de un Servicio síncrono

Un servicio de mensajes o llamadas síncrono proporciona una respuesta inmediata a una consulta. Un proceso BPEL puede realizar una conexión a un servicio web con estas características, a través del *PartnerLink*, recibiendo una respuesta de manera sincrónica. La actividad `<invoke>`, que establece la conexión con el *PartnerLink*, crea un único puerto en el proceso BPEL para enviar los datos y recibir la respuesta. (Matjaz Juric, 2010)

Para el control de fallas, cada dominio tiene un atributo *syncMaxWaitTime*, que tiene un valor por defecto de 60 segundos, el cual puede ser configurado por el administrador. Si el proceso BPEL no recibe la respuesta en el tiempo determinado, la actividad fallará. En la siguiente figura se puede apreciar el proceso BPEL síncrono. (Matjaz Juric, 2010)

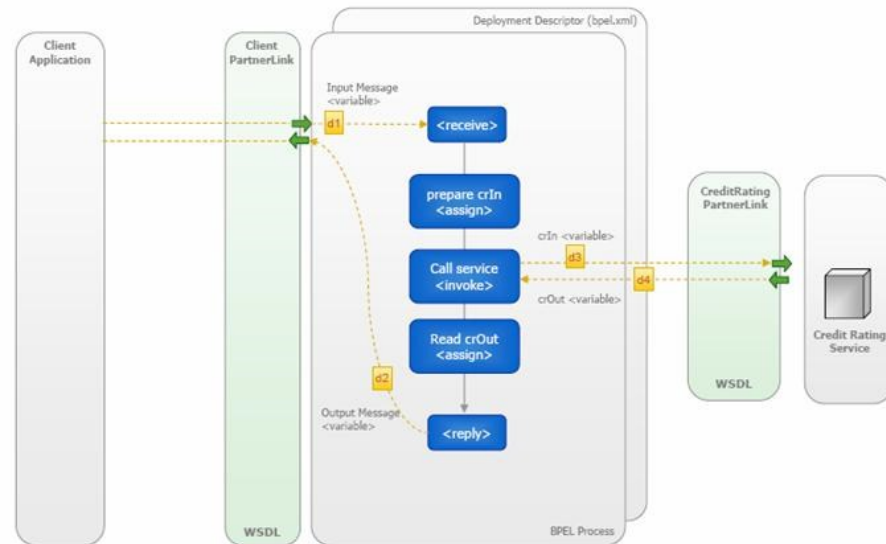


Figura 2.13 Invocación de un Servicio Síncrono. (Matjaz Juric, 2010)

### Invocación de un servicio asíncrono

Un servicio asíncrono en cambio tiene más utilidad en aquellas situaciones en donde un servicio tiene un tiempo considerable para responder al cliente. Por lo tanto, en estas situaciones se utiliza una actividad que llame al servicio web y otro que espere su respuesta sin límite de tiempo. Para realizar esto, los datos se envían en una variable, junto a un identificador particular para el cliente y el *PartnerLink*, lo que asegura que la respuesta que llega del *PartnerLink*, le llegue al solicitante.

BPEL utiliza la actividad *<receive>* para esperar la respuesta del servicio Web, la cual es almacenada en una variable de respuesta. Un proceso BPEL asíncrono utiliza dos *PortTypes*: la operación *<initiate>* de la actividad *<invoke>* y la operación *<onResult>* de la actividad *<receive.>* La siguiente figura visualiza el esquema básico de un proceso BPEL asíncrono. (Matjaz Juric, 2010)



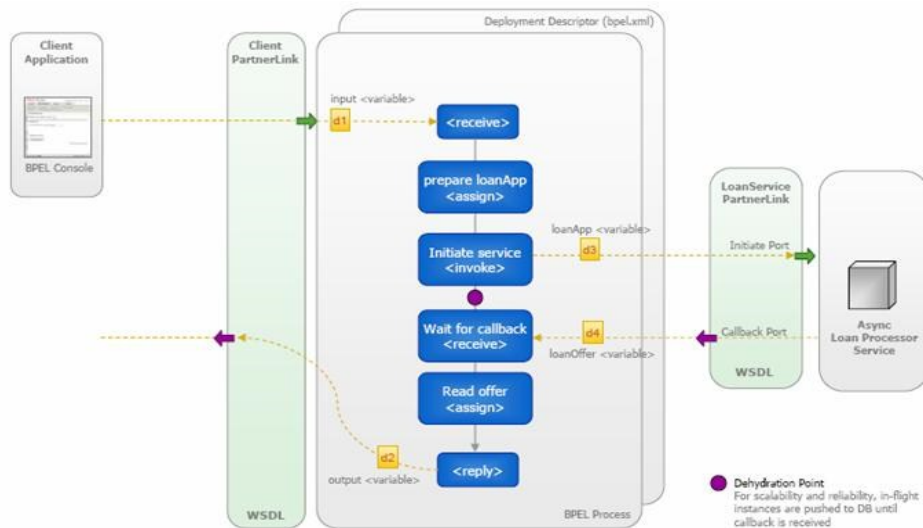


Figura 2.14 Invocación de un Servicio Asíncrono. (Matjaz Juric, 2010)

### 2.1.6 Gestión de la comunicación en los Procesos BPEL

La infraestructura de comunicación que provee los mensajes internos, es capaz de conectar los componentes y habilitar el flujo de datos, para eso esta infraestructura tiene que:

- Recibir mensajes de los proveedores de servicios o socios externos mediante el protocolo de comunicación SOAP, adaptadores, o a través de API en forma de XML.
- Encaminar a los mensajes basados en la definición de compuestos, para el respectivo motor de servicios.

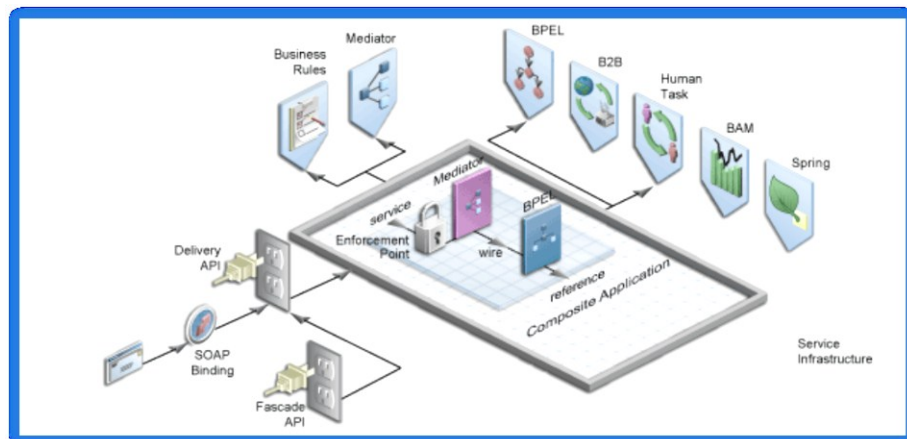


Figura 2.15 Comunicación de los Servicios BPEL. (Oracle, 2010)

### **Mediador de Oracle:**

El Mediador de Oracle y el servicio de infraestructura ofrecen las siguientes combinaciones de capacidades: **(Beecher, 2011)**

- Enrutar los servicios para proporcionar el movimiento de datos.
- Encaminar las reglas para especificar las rutas, documentos de transformación y refinamiento de datos.
- Suscripción a los eventos de negocios.
- Unificación del manejo de errores y la gestión de excepciones.

El Mediador de Oracle encamina los datos de los proveedores de servicios a los *host* externos, correspondiente a una carga equilibrada de enrutamiento de tráfico HTTP.

El mediador de Oracle es un componente clave para comunicarse con el bus de servicios empresariales, desde su punto de entrada hasta su punto de salida. Esto lo hace a través de reglas, las cuales determinan como una instancia de mensajes procesados llegan a su punto de destino. Por lo tanto, el mediador de Oracle puede utilizar estas reglas para realizar las siguientes acciones. **(Beecher, 2011)**

*Route* (Encaminar): Determina los componentes de servicios (procesos BPEL, reglas de negocio, tareas humanas y mediador), a la cual envía los mensajes.

*Validate* (validador): Proporciona el soporte necesario para validar los inconvenientes en la carga de mensajes por usar un *Schematron* o un archivo XSD.

*Filter* (filtro): Se aplica para filtrar expresiones que especifican los contenidos (carga útil) de un mensaje para ser analizado, antes de cualquier servicio que sea invocado.

Transformación: Transforma los datos de un documento esquema XML a otro, de este modo habilita el intercambio de información entre las aplicaciones usadas en diferentes esquemas.

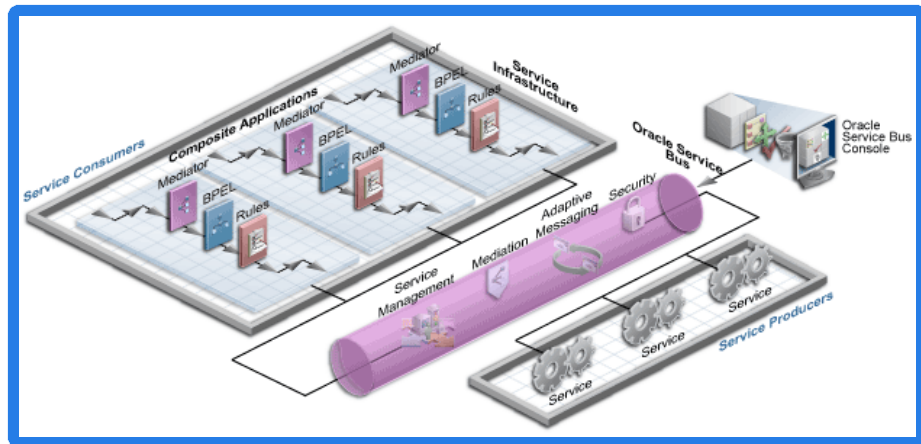


Figura 2.16 Servicios del Bus de Aplicaciones de Oracle. (Oracle, 2010)

### Bus de Servicios Oracle:

El Bus de Servicios Oracle, es un intermediario que procesa los mensajes entrantes de una solicitud de servicio, determinando una lógica de enrutamiento para transformar estos mensajes. Esto hace que sean compatibles con los consumidores de servicios. Además, recibe mensajes a través de un protocolo de transporte como HTTP(S), JMS, Archivos y FTP, y envía mensajes a través de la misma o de un protocolo de transporte diferente. Estos servicios de respuesta de mensajes siguen una ruta inversa. (Oracle, 2009).

El Bus de Servicios de Oracle se conecta, interacciona y gestiona las comunicaciones entre servicios heterogéneos, no solo servicios web, sino también Java, .NET, servicios de red, servicios de mensajería y los puntos de legado. Esto extraordinariamente entrega la capacidad de integración de un Bus de Servicios Empresarial (ESB) con la administración de servicios operativos, las cuales exponen las siguientes áreas funcionales: (Oracle, 2009)

- **Gestión:** Proporciona las capacidades integradas de administración de los servicios, dando un manejo optimizado de todos los mensajes. El soporte de prevención, asegura que los procesos críticos de negocios continúen satisfaciendo a las necesidades de los clientes, tales como: las demandas del negocio, requisitos, los cambios de cargas de trabajo, etc.
- **Mediación:** Ofrece un ambiente para el enrutamiento basado en contenidos, las transformaciones de mensajes y las orquestaciones simples.
- **Adaptación de mensajería:** Conecta de forma fiable cualquier servicio, mediante un sistema de transporte de servicios web estándar, protocolos de mensajería y la configuración de la demanda específica del uso de transporte de servicios.
- **Seguridad:** Tiene un servicio rápido de configuración y entorno de integración que son asociadas a políticas abstractas con reglas, seguridades, y servicios de acceso de punto-destino.

## **2.2 Adaptadores: Tecnología y Aplicaciones**

### **2.2.1 Introducción**

Para que los Sistemas Empresariales se integren con otros sistemas, ya sean tecnológicos o aplicaciones empaquetadas, se requiere de un *framework* adaptador que soporte las conexiones a varios sistemas o servicios. Los adaptadores de Oracle utilizan la tecnología JCA para conectar los sistemas externos con SOA.

### **2.2.2 Tecnologías Relacionadas**

Las Tecnologías relacionadas con los adaptadores se ostentan en las actividades *<invoke>*, *<receive>*, *<reply>*, y en la sección de mensajes de selección de actividades. Por lo tanto, se debe crear las variables de cabecera para ser usadas con los adaptadores que se describen a continuación: **(Oracle, 2010)**

- *Advanced Queuing (AQ)*: Para la interacción con una cola de espera. AQ provee un mecanismo flexible de forma bidireccional y comunicación asíncrona entre las aplicaciones participantes.
- *Business Activity Monitoring (BAM)*: Para información de datos a los objeto de datos en un servidor de Oracle BAM.
- FTP y Archivos: Para la modificación de archivos (lectura y escritura) en un sistema de archivos locales y en sistema de archivos remoto (para el uso del protocolo de transferencia de archivos o en sus siglas (FTP)).
- *Java Messaging Service (JMS)*: Para la interacción con JMS. La arquitectura JMS es usada en una interfaz cliente para muchas arquitecturas de servidores de mensajería.
- *Message Queue (MQ)*: Para modificar los mensajes con el sistema de colas de espera *WebSphere MQ*.

### 2.2.3 Aplicaciones Relacionadas

- PeopleSoft: *PeopleSoft Enterprise* de Oracle se ha desarrollado para tratar los requisitos más complejos en los procesos de negocios. Permiten a las empresas mejorar su funcionamiento, ya que ofrecen soluciones más comprensivas de los procesos de negocios y de las industrias. Soporta una amplia infraestructura tecnológica y un entorno heterogéneo de usos, en donde se pueden integrar los servicios web a las aplicaciones de PeopleSoft Enterprise. Esto lo puede realizar mediante una configuración simple, la cual asegura de que los requisitos más particulares del cliente puedan ser compensados. **(Vasudevan, 2006)**

- SAP R/3: Permite controlar todos los procesos a través de módulos, en la cual los módulos se integran y forman un sistema de gestión de procesos para las empresas. También, el entorno de desarrollo de SAP y sus sistemas de información, proporciona a los clientes herramientas para el desarrollo, integrando a los sistemas los requerimientos particulares. SAP R/3, ofrece a los usuarios un lenguaje de comunicación propio de cuarta generación llamado ABAP/4, construido para las necesidades del negocio. Por lo tanto, esta aplicación es altamente modular, de manera que facilita el control de la interacción entre cliente/servidor aplicado a diversos niveles.
- Siebel: Es una aplicación interoperable de negocios, definida como sistema Siebel, que a su vez engloba a varias aplicaciones diseñadas para la gestión de los negocios. *Siebel System*, se relaciona con ERM, CRM, PRM, que son aplicaciones diseñadas para automatizar los aspectos de negocios, permitiéndoles a las organizaciones desenvolverse y coordinarse de forma asociada. **(Vasudevan, 2006)**
- *JD Edwards OneWorld*: Es una aplicación E.R.P que a su vez se relaciona con *PeopleSoft* y termina complementándose con *OneWorld* que está alineado a *PeopleSoft*. **(Vasudevan, 2006)**

#### **2.2.4 Tipos de Servicios de Adaptadores**

Los servicios adaptadores soportan múltiples plataformas y marcas, por lo que les permite acceder a mainframe, aplicaciones legacy, telecomunicaciones, etc.

Los adaptadores de Oracle *Application Server* permiten distribuir o integrar fácilmente los sistemas empresariales con otros servicios. Estos adaptadores se describen en las siguientes categorías. **(May, 2012)**.

**Adaptadores de Tecnología:** Los adaptadores de tecnología se integran a los siguientes protocolos de transporte, almacenes de datos, y mensajería middleware: **(May, 2012)**

- BAM
- FTP
- Java Messaging Service (JMS)
- Advanced Queuing (AQ)
- Files
- Message Queuing (MQ) Series

**Adaptadores de Aplicación Empaquetados:** Se integran con SOA y otros paquetes de aplicaciones como SAP, Siebel y que incluyen también: **(May, 2012)**

- Oracle Applications
- PeopleSoft
- SAP R/3
- Siebel
- JD Edwards OneWorld

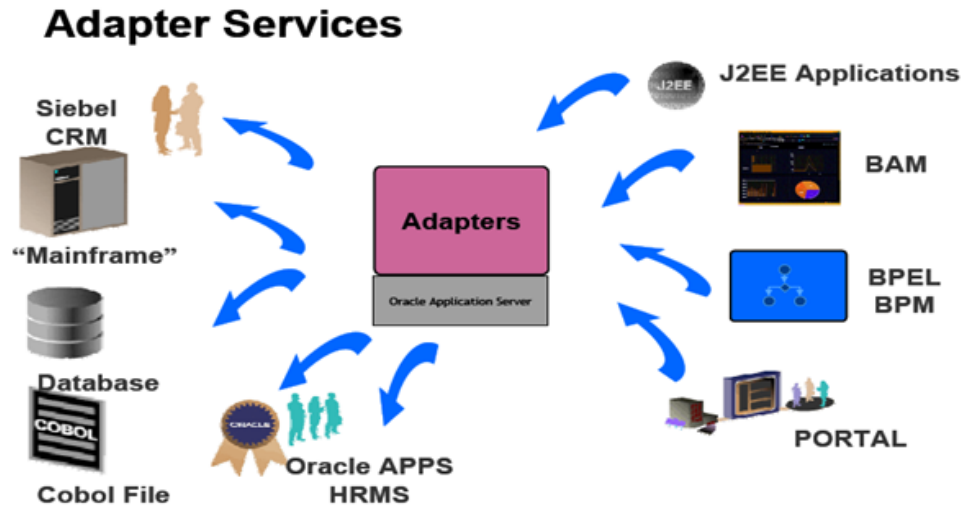


Figura 2.17 Servicios de Adaptadores (May, 2012)

La arquitectura orientada a servicios (SOA) puede ser desplegada por Oracle *Application Server*, ya que es parte de Oracle *Fusion Middleware*, declarándose como una plataforma de aplicaciones (APS), ofreciendo a la vez una solución para desarrollar, integrar y desplegar aplicaciones empresariales, y portales de sitios web. Permite también desplegar sistemas modulares abiertos, facilitando la adaptación de sistemas existentes con diferentes aplicaciones empresariales.

## 2.3 Flujos de Trabajo Humano

### 2.3.1 Introducción

Los procesos BPEL pueden realizar operaciones más complejas como por ejemplo actualizar, completar, reanudar, alinear y subir tareas. Para eso los procesos BPEL tienen las tareas asignadas a usuarios, los cuales tienen la capacidad de interactuar con ellas, usando las aplicaciones más convenientes. Las aplicaciones se comunican con los servicios de flujos de trabajo, usando las interfaces WSDL o APIs tales como Java, las cuales permiten adquirir la lista de tareas para el usuario; interpretando la interfaz usuario apropiada y devolviendo los resultados a los servicios de flujos de trabajo.



Un servicio de flujo de trabajo podría ser e-mails y mensajes de texto (SMS), los cuales están habilitados para interactuar con los procesos BPEL de una manera relativamente fácil. Los componentes de *Human Workflow* se caracterizan por prestar diferentes servicios que engloban varios aspectos de interacción humana con los procesos de negocio y a la vez exponen sus interfaces a WSDL; por consiguiente, el proceso BPEL los invoca tan pronto como cualquier otro servicio. (Matjaz Juric, 2010)

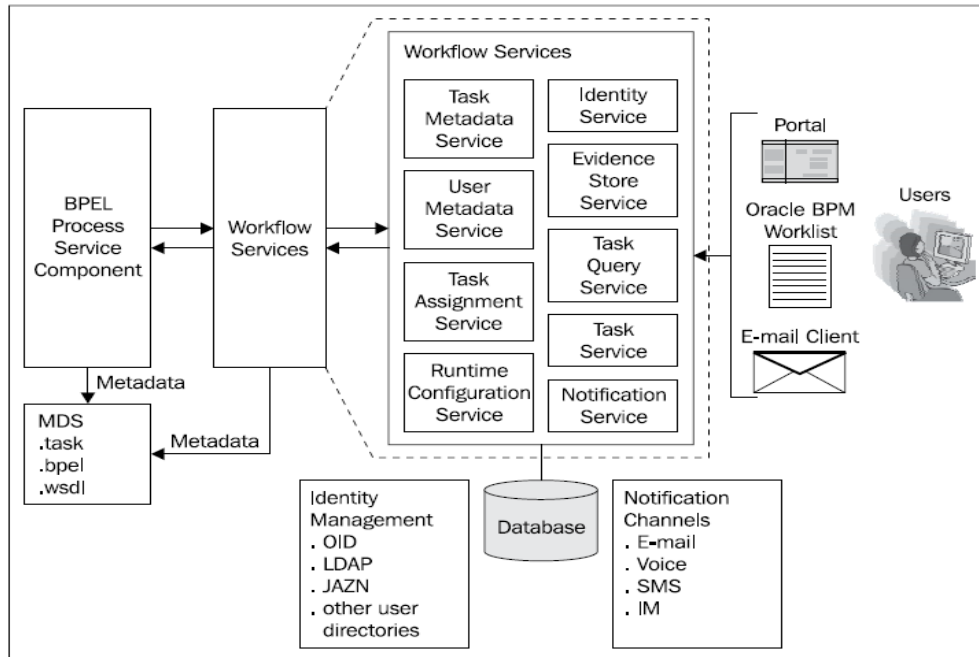


Figura 2.18 Human Workflow. (Matjaz Juric, 2010)

### 2.3.2 Especificación de Requerimientos de Flujo de Trabajo Humano

Los requerimientos en un flujo de trabajo humano son: (Matjaz Juric, 2010)

- Accesos de Seguridad: Esto incluye el manejo de seguridad, identidad que se le provee a los usuarios.
- Asignación de Tareas: Es la lista de tareas que surgen en un flujo de trabajo, las cuales deben realizarse junto con el usuario seleccionado para dichas tareas.
- Escalamiento o montaje de acciones y las reglas de delegación.

- Asignación de lista de tareas personalizables para el usuario.
- Normas para el alineamiento o enrutamiento de tareas en conjunto con la creación de patrones en el flujo de trabajo y en las reglas de trabajo.
- Auditoria de la lista de tareas creada y un historial de las mismas, en donde se incluirán los reportes de productividad.

### 2.3.3 Gestión de los Servicios en el Flujo de Trabajo de Oracle BPEL

Los servicios a gestionar por Oracle BPEL son los siguientes: **(Matjaz Juric, 2010)**

- Servicios de Tarea(*Task Service*): Este tipo de servicio, expone las operaciones para gestionar el estado de las tareas, en donde una operación puede estar actualizando una tarea, completando una tarea, reasignando una tarea, etc. Cuando se asigna una tarea humana a los procesos BPEL, los *PartnerLink* correspondientes para los servicios de tarea son automáticamente creados.
- Servicio de asignación de tareas: Provee la funcionalidad de alinear, montar, reasignar tareas y más.
- Servicio de consulta de tareas: Habilita el restablecimiento de la lista de tareas para un usuario, basándose en criterios de búsqueda.
- Servicio de metadatos de tareas: Restablece los metadatos de las tareas.
- Servicio de Identidad: Proporciona la autenticación y autorización de usuarios y revisa las propiedades y privilegios que tiene el usuario.
- Servicio de Notificación: Habilita el envío de notificaciones para usuarios, usando varios canales como (e-mail, mensajes de voz, IM, SMS, and so on).

- Servicio de metadatos de usuario: Administra los metadatos, relacionado con los flujos de trabajo de usuarios, así como colas de trabajo de los usuarios, preferencias, etc.
- Servicio de ejecución de configuración: Provee la funcionalidad para la gestión de los metadatos usados en los servicios de tareas en un ambiente de ejecución.
- Servicio de almacenamiento de pruebas: Administra el soporte de señal-digital del flujo de trabajo de las tareas.

Como conclusión, se puede decir que BPEL es el motor principal para ejecutar los procesos de negocios en una orquestación de servicios, ya que permite definir el orden exacto en cada servicio compuesto. Además, BPEL puede invocar los servicios sincrónicamente y asincrónicamente, describiendo los roles de cada WSDL. Para comunicarse con los servicios web, BPEL debe hacerlo mediante los *PartnerLink* y las actividades de interacción de datos, ya que son necesarias para invocar operaciones, enviar mensajes y recibir las respuestas para sus clientes o servicios externos.

## CAPITULO 3

### ANALISIS Y DISEÑO

Es este capítulo se determinará el análisis y diseño del sistema de solicitudes de crédito, por lo que se describen los diagramas necesarios que identifican cada acción que se dé al realizar una solicitud de crédito comercial. También, se detallarán los diseños de las interfaces que serán implementadas para construir la aplicación de demostración de una orquestación de servicios web.

#### **3.1 Definición General del Sistema**

El sistema consta de una aplicación basada en reglas de negocios, la cual permitirá tramitar el proceso de una solicitud de crédito comercial de una manera más ágil. Está diseñado para demostrar que la aplicación de Oracle BPEL es una solución completa para orquestar distintos sistemas desarrollados en distintos lenguajes de programación y con diferentes bases de datos.

#### **3.2 Análisis de Requerimientos de solicitudes de Crédito**

Para proceder a desarrollar el sistema, se tuvo que realizar las entrevistas respectivas en distintos almacenes que dan crédito comercial a sus clientes. En estas entrevistas se confirmó, que para requerir una solicitud de crédito comercial y por ende ser aprobada, el cliente debe proporcionar sus datos personales, la estabilidad laboral, información de la ubicación del domicilio y un recibo de luz, agua o teléfono.

Por lo tanto, la información que requiere el representante de ventas para aprobar el crédito solicitado; se simulará en tres sistemas, los cuales serán diseñados en distintos lenguajes de programación y con distintas bases de datos.

El sistema que se desarrollará para simular la entidad del **registro civil** será hecho en un entorno web desarrollado en PHP y a la vez en una base de datos MySQL, el sistemas que se desarrollará para simular la entidad del **registro de la propiedad** será desarrollado en .NET y los datos serán guardados en una base de datos SQLServer 2008. Por último, el registro de ventas de los clientes que han solicitado un crédito comercial serán visualizados en una aplicación creada en ADF Business Component y por ende la información que se muestre será registrada en una base de datos Oracle.

### 3.2.1 Diagramas de Casos de Uso de solicitudes de Crédito

A continuación se presenta el flujo de eventos de los casos de uso de una Aprobación de Crédito.

#### Diagramas de Casos de Uso de la Aprobación del Crédito.

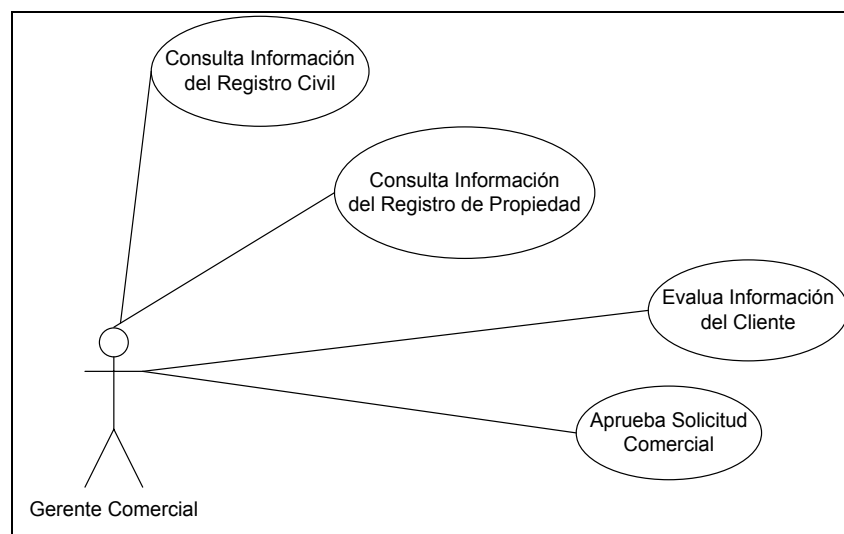


Figura 3.1 Aprobación de crédito

#### Flujo de eventos del Caso de Uso Consulta información Registro Civil.

<b>Nombre : Consulta Información Registro Civil</b>	
<b>Descripción</b>	El sistema debe mostrar la información del cliente obtenida de la base de datos del Registro Civil.
<b>Entrada</b>	El Gerente de Ventas debe ingresar el número de cedula del Cliente para poder consultar sus datos personales.
<b>Proceso</b>	Consulta los datos personales del Cliente que se encuentran almacenados en la base de datos del Registro civil.
<b>Salida</b>	Visualiza la información principal del Cliente.

Tabla 3.1 Registro Civil

### Flujo de eventos del Caso de Uso Consulta información Registro Propiedad.

<b>Nombre : Consulta Información Registro Propiedad</b>	
<b>Descripción</b>	El sistema debe mostrar la información del cliente obtenida de la base de datos del Registro de la Propiedad.
<b>Entrada</b>	El Gerente de Ventas debe ingresar el número de cedula del Cliente para poder consultar los datos de su propiedad.
<b>Proceso</b>	Consulta los datos de la Propiedad del Cliente que se encuentran almacenados en la base de datos del Registro de la Propiedad.
<b>Salida</b>	Visualiza la información principal de la Propiedad del Cliente.

Tabla 3.2 Registro de Propiedad

### Flujo de eventos del Caso de Uso Evalúa Información del Cliente.

<b>Nombre : Evalúa Información del Cliente</b>	
<b>Descripción</b>	El sistema evaluara si la información dada por el cliente es correcta o incorrecta.
<b>Entrada</b>	Si los datos del cliente son incorrectos, el Gerente de Ventas deberá ingresar nuevamente un numero de cedula válido.
<b>Proceso</b>	Valida los datos proporcionados por el cliente, comparándolos con los datos personales que se encuentran almacenados en la base de datos del Registro civil y del Registro de la Propiedad.
<b>Salida</b>	Visualiza la información correcta del Cliente.

Tabla 3.3 Información del Cliente

### Flujo de eventos del Caso de Uso Aprueba Solicitud Comercial.

Nombre :Aprueba Solicitud Comercial	
<b>Descripción</b>	El sistema mostrara un mensaje de crédito aprobado si toda la información es correcta, en caso contrario emitirá un mensaje de crédito no aprobado.
<b>Entrada</b>	El Gerente de Ventas debe ingresar el número de cedula del Cliente para poder consultar sus datos personales verificar si son válidos para poder aprobar el crédito.
<b>Proceso</b>	Evalúa toda la información del cliente y decide si se aprueba o no el crédito.
<b>Salida</b>	Visualiza un mensaje de aprobación o no aprobación de la solicitud de crédito.

Tabla 3.4 Solicitud Comercial

### Diagramas de Casos de Uso de la Generación de Venta.

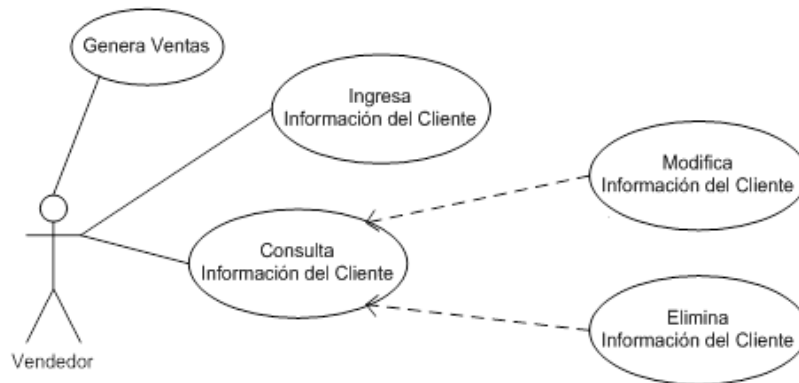


Figura 3.2 Generación de Venta

### Flujo de eventos del Caso de Uso Genera Ventas.

Nombre : Genera Ventas	
<b>Descripción</b>	El sistema debe guardar el monto total, el número de factura y la fecha en la que el cliente adquirió los productos.
<b>Entrada</b>	El vendedor se encargara de ingresar el valor total, el número de factura y la fecha de los productos adquiridos por el cliente.
<b>Proceso</b>	Validar si el valor total, el número de factura y la fecha ingresados sean correctos.
<b>Salida</b>	Visualiza un mensaje en el caso de que la información ingresada sea incorrecta.

Tabla 3.5 Genera Ventas

### Flujo de eventos del Caso de Uso Ingresa Información del Cliente.

<b>Nombre : Ingresa Información del Cliente</b>	
<b>Descripción</b>	El sistema debe guardar la información del cliente.
<b>Entrada</b>	El vendedor deberá ingresar los datos personales del cliente que solicita el crédito comercial.
<b>Proceso</b>	Almacena los datos correctos de un cliente en una base de datos.
<b>Salida</b>	Despliega un mensaje que la información a sido almacenada correctamente.

Tabla 3.6 Ingresar Cliente

### Flujo de eventos del Caso de Uso Consulta Información del Cliente.

<b>Nombre : Consulta Información del Cliente</b>	
<b>Descripción</b>	El sistema consulta la información de un cliente que se encuentre almacenado en la base de datos de la Empresa Comercial.
<b>Entrada</b>	El vendedor debe ingresar el número de cedula para poder consultar los datos del cliente.
<b>Proceso</b>	Realiza una búsqueda en la base de datos de la Empresa comercial mediante el número de cedula del cliente.
<b>Salida</b>	Visualiza la información principal del Cliente en caso de encontrarlo, de lo contrario emitirá un mensaje indicando que no se ha encontrado el cliente.

Tabla 3.7 Consulta Cliente

### Flujo de eventos del Caso de Uso Modifica Información del Cliente.

<b>Nombre :Modifica Información del Cliente</b>	
<b>Descripción</b>	El sistema permite realizar los cambios en la información de un cliente que este creado en la base de datos.
<b>Entrada</b>	El vendedor ingresara la información respectiva en los campos que pretendan ser modificados.
<b>Proceso</b>	Modifica la información de uno o varios clientes que hayan sido almacenados en la base de datos.
<b>Salida</b>	Visualiza un mensaje indicando que los datos se han modificado exitosamente.

Tabla 3.8 Modifica Cliente



**Flujo de eventos del Caso de Uso Elimina Información del Cliente.**

<b>Nombre : Elimina Información del Cliente</b>	
<b>Descripción</b>	El sistema permite eliminar uno o más clientes almacenados en la base de datos.
<b>Entrada</b>	El vendedor ingresara la cedula del cliente a ser eliminado.
<b>Proceso</b>	Eliminar a uno o varios clientes que haya sido creados anteriormente en la base de datos.
<b>Salida</b>	Visualiza un mensaje indicando que el cliente a sido eliminado.

Tabla 3.9 Elimina Cliente

**Diagramas de Casos de Uso del Registro Civil.**

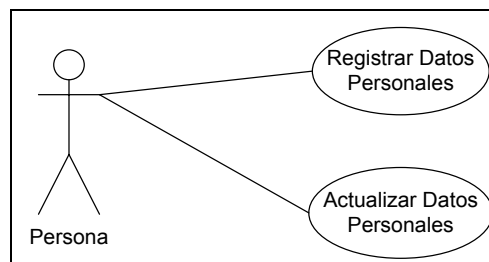


Figura 3.3 Registro Civil

**Flujo de eventos del Caso de Uso Registrar Datos Personales.**

<b>Nombre : Registrar Datos Personales</b>	
<b>Descripción</b>	El sistema guarda los datos personales proporcionados por una persona.
<b>Entrada</b>	El usuario del sistema debe ingresar todos los datos personales de una persona.
<b>Proceso</b>	Guarda los datos personales de una persona en base de datos del registro civil.
<b>Salida</b>	Visualiza un mensaje indicando que la información se ha guardado correctamente.

Tabla 3.10 Registra Persona

## Diagramas de Casos de Uso del Registro de Propiedad.

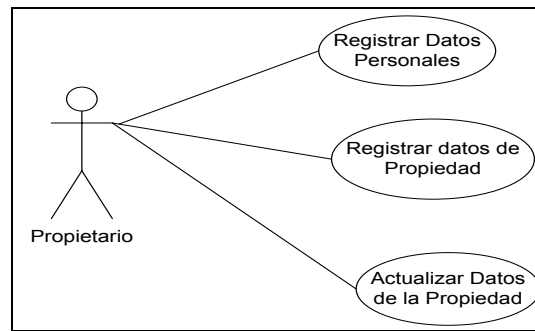


Figura 3.4 Registra Propiedad

### Flujo de eventos del Caso de Uso Registrar Datos Personales.

Nombre : Registrar Datos Personales	
<b>Descripción</b>	El sistema registrara los datos personales del propietario.
<b>Entrada</b>	El usuario del sistema ingresara los datos personales del propietario.
<b>Proceso</b>	Guardar los datos personales en la base de datos del registro de la propiedad.
<b>Salida</b>	Visualiza un mensaje que la información a sido registrada con éxito.

Tabla 3.10 Registra Persona

### Flujo de eventos del Caso de Uso Registrar Datos de la Propiedad.

Nombre :Registrar datos de la Propiedad	
<b>Descripción</b>	El sistema registrara los datos de la propiedad.
<b>Entrada</b>	El usuario del sistema deberá ingresar los datos de la propiedad proporcionados por el propietario.
<b>Proceso</b>	Guardar los datos de la propiedad en una base de datos del registro de la propiedad.
<b>Salida</b>	Visualiza un mensaje que la información a sido registrada con éxito.

Tabla 3.11 Registra Propiedad

### Flujo de eventos del Caso de Uso Actualizar Datos de la Propiedad.

<b>Nombre :Actualizar datos de la propiedad</b>	
<b>Descripción</b>	El sistema permite actualizar uno o más datos del propietario o de la propiedad que hayan sido almacenados en la base de datos.
<b>Entrada</b>	El usuario del sistema deberá ingresar los datos que vayan a ser actualizados.
<b>Proceso</b>	Modificar los datos tanto del propietario como de la propiedad que hayan sido creados anteriormente en la base de datos del registro de la propiedad.
<b>Salida</b>	Visualiza un mensaje indicando que los datos han sido actualizados correctamente.

Tabla 3.12 Modifica Propiedad

### 3.2.2 Diagrama de Clases de las solicitudes de Crédito

A continuación se presenta el diagrama de clases de una Solicitud de Crédito Comercial.

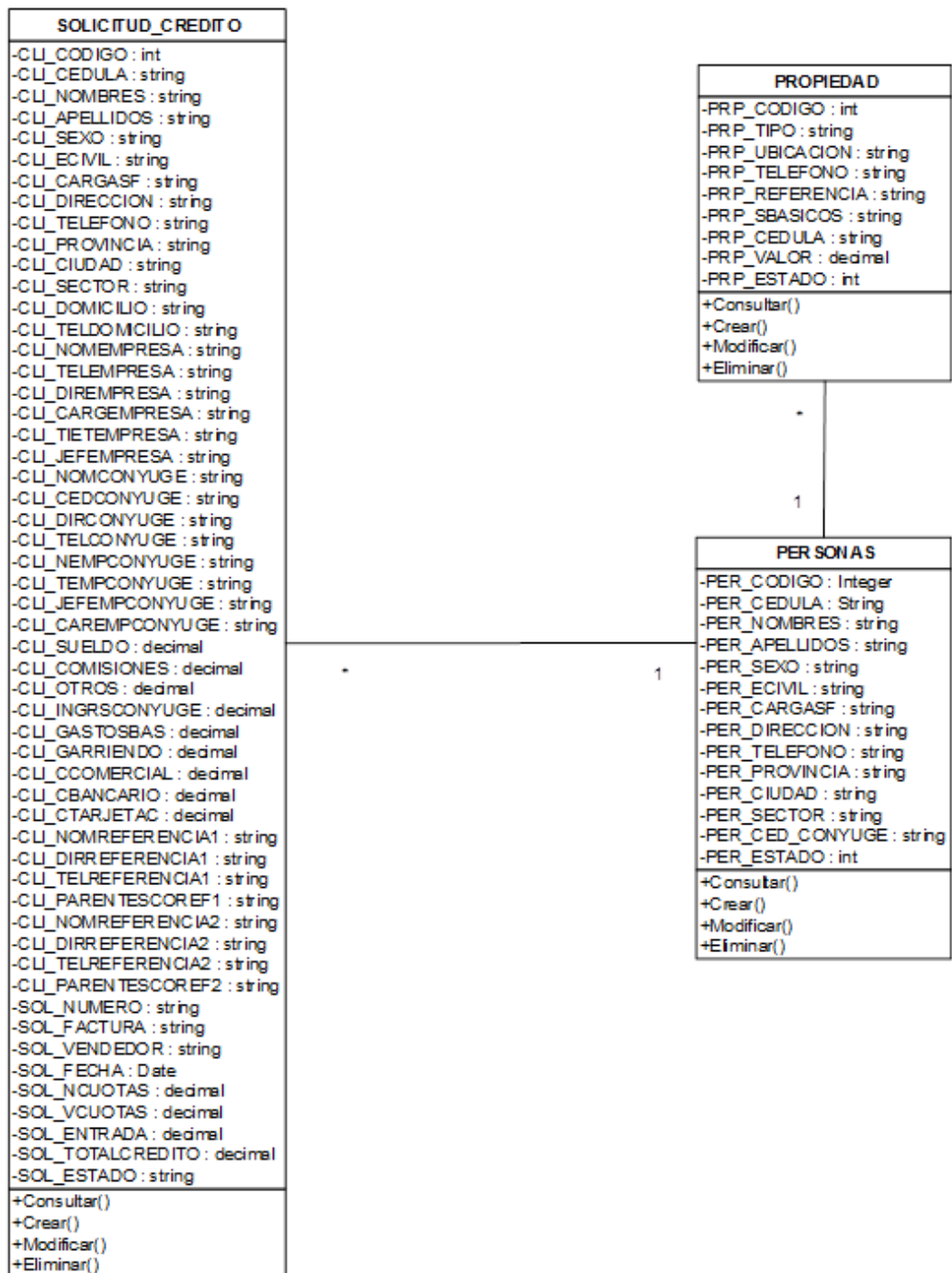


Figura 3.5 Diagrama Entidad-Relación Solicitudes de Crédito

### 3.3 Modelo Conceptual de solicitudes de Crédito

A continuación se presenta el modelo conceptual de una Solicitud de Crédito Comercial.

Este diagrama ayuda a entender cómo interactúan los componentes en el momento en que se realiza una solicitud de crédito comercial.

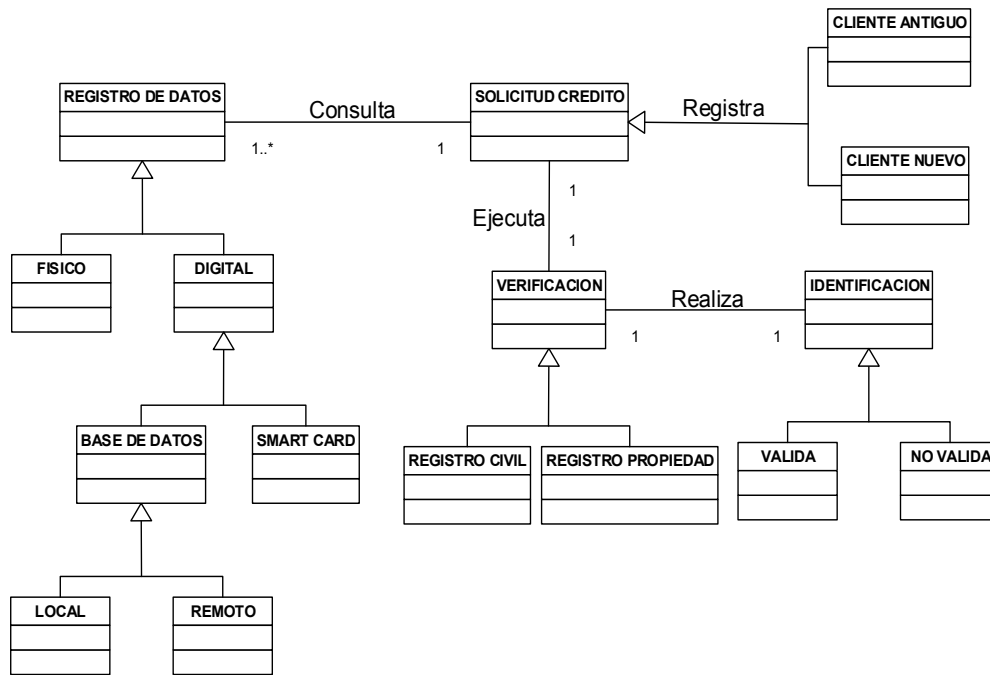


Figura 3.6 Diagrama Conceptual Solicitud de Crédito

### 3.4 Diagrama de Componentes de solicitudes de Crédito

A continuación se presenta el diagrama de Componentes de una Solicitud de Crédito Comercial.

El primer Diagrama de componentes representa al Sistema Global, en la cual se describe su razón de dependencia de cada uno de los componentes.

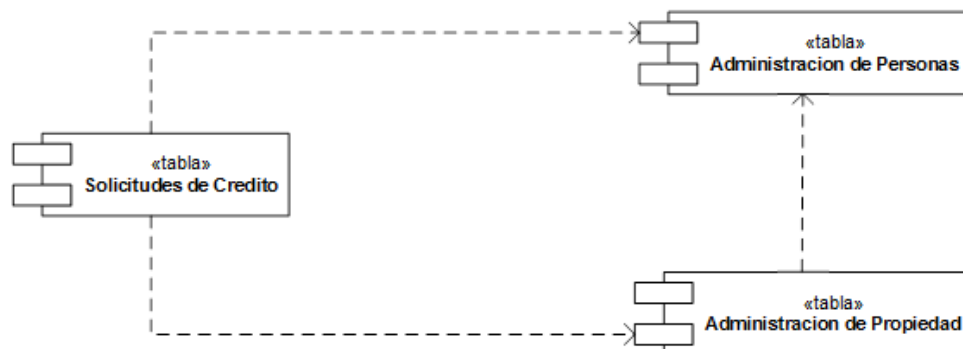


Figura 3.7 Diagrama de Componentes Solicitudes de Crédito

El segundo diagrama hace referencia al diagrama de componentes de Administración de Personas, en la cual se muestra su razón de dependencia entre cada una de sus acciones ejecutables.

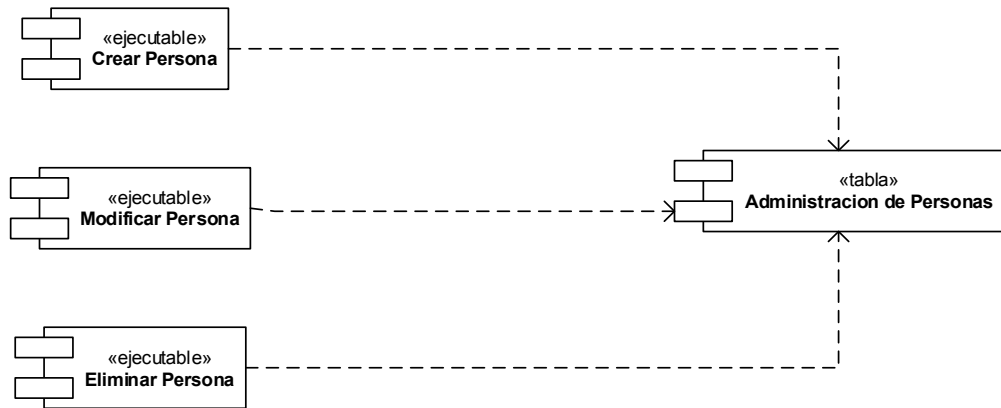


Figura 3.8 Diagrama de Componentes Administración de Personas

El tercer diagrama hace referencia al diagrama de componentes de Administración de Propiedad, en la cual se especifica su razón de dependencia entre cada una de sus acciones ejecutables.

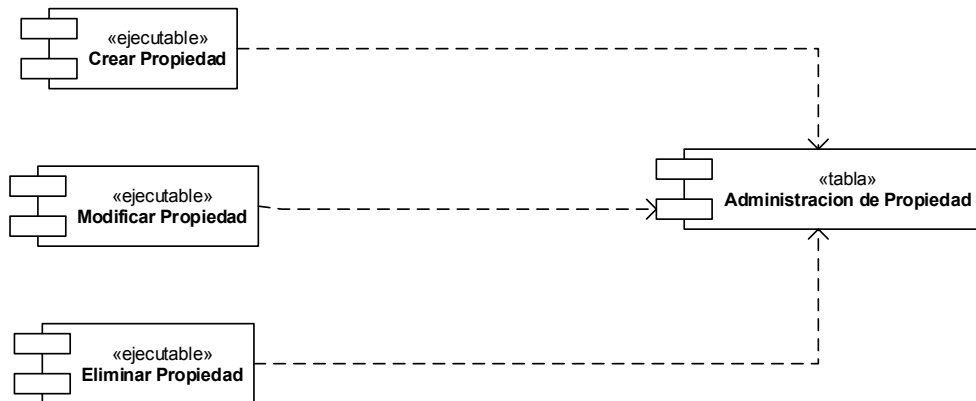


Figura 3.9 Diagrama de Componentes Administración de Propiedad

### **3.5 Diseño de la Arquitectura para el Sistema de Solicitudes de Crédito**

EL siguiente diseño describe la arquitectura del sistema de Solicitudes de Crédito Comercial, el cual consta de un portal web para que el vendedor pueda realizar las consultas necesarias de los datos personales del cliente que requiera un crédito comercial. Cada solicitud enviada por el vendedor mediante el portal web, es procesada por el motor de procesos de negocios BPEL, mediante la identificación de variables de entrada que están dentro de los esquemas expuestos por los archivos WSDL. Una vez que se establecen los vínculos entre los *Web Services* y BPEL, este se encarga de orquestrar las solicitudes mediante un flujo de actividades.

Luego de que BPEL procesa la solicitud, esta es expuesta a otros archivos WSDL que contienen las variables de salida que cumplirán la función de obtener la información solicitada por el usuario. Esta información se consulta en las distintas bases de datos que almacenan la información personal del cliente. Este proceso se llama mapeo de datos.

Por ejemplo, en el caso de solicitar un crédito comercial, primero el vendedor deberá pedirle el número de cédula al cliente, este número va a ser ingresado en el portal de solicitudes de crédito y luego ejecutará la acción de consulta. La respuesta que debe visualizarse, serán todos los datos personales del cliente y también los datos de su propiedad.

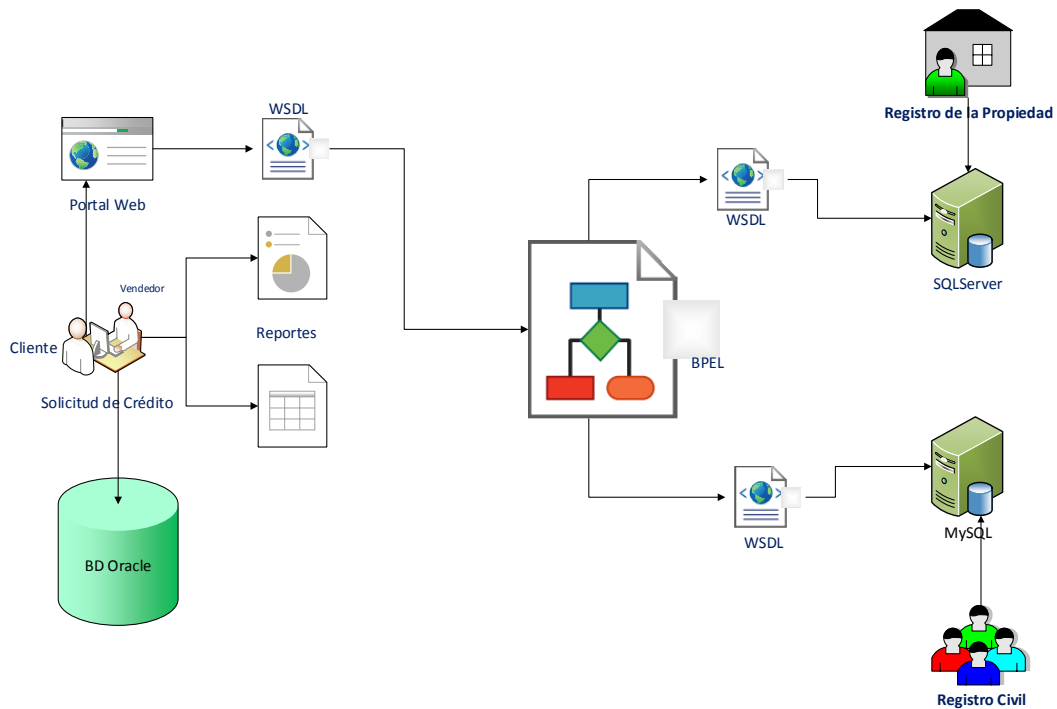


Figura 3.10 Arquitectura Solicitud de Crédito Comercial

### 3.6 Procesos de Interacción del Sistema de solicitudes de Crédito

Analizando el sistema se puede observar que la solicitud que el cliente envía, es por medio de un mensaje WSDL. Este mensaje llega a la Actividad *Receive*, que es la actividad principal de un proceso BPEL, y está a la espera de los mensajes que pueda enviar el cliente. Luego, el dato que fue recibido por la actividad *receive*, es mapeada con las variables de salida de la actividad *invoke*, la cual invocará a un servicio web síncrono de solicitud/respuesta que es identificado por un *partnerlink*.

Este *partnetlink* especificará una operación de consulta de personas casadas que el servicio web debe realizar en la base de datos del Registro Civil enlazada.



Por lo tanto, una vez realizadas las operaciones por parte del *web services*, este recibirá una respuesta inmediata de la información solicitada en la variable de salida y nuevamente estas variables serán mapeadas en la siguiente actividad *invoke*. Esta actividad en cambio le dirá al *partnerlink* que realice la operación de consulta de propiedad, en la base de datos del Registro de la Propiedad enlazada, de acuerdo a la información obtenida en la primera consulta.

Una vez que se obtiene la información de la propiedad del cliente en la variable de salida del servicio web, este realizará una serie de validaciones con los resultados y si todo está bien, mapeará toda la información obtenida a las variables de entrada de la siguiente actividad *invoke*. En consecuencia, el mensaje de respuesta se enviará de la actividad *reply*, que es la actividad que permite enviar un mensaje en respuesta, a otro mensaje recibido a través de una actividad *recv*. Este mensaje sería Aprobado en el caso de que toda la información este completa y sea válida, en caso contrario dará una respuesta de Rechazado y el cliente no podrá realizar un crédito comercial.

El siguiente diagrama describe la interacción de los procesos en el sistema de orquestación de una Solicitud de Crédito.

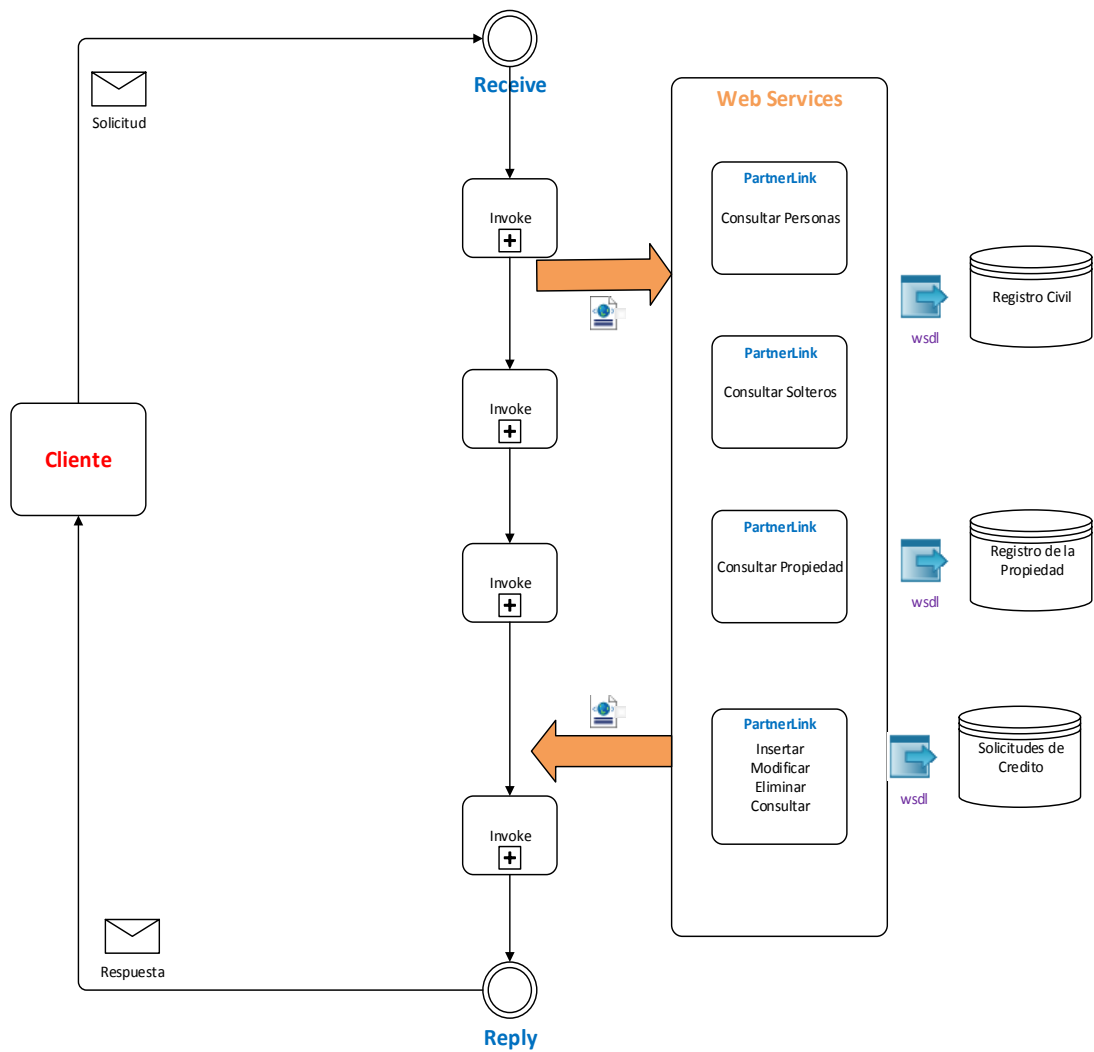


Figura 3.11 Procesos de Interacción de una Solicitud de Crédito

### 3.7 Diseño de la Solución

#### 3.7.1 Plataforma Tecnológica

Define las tecnologías que se utilizarán para la implementación y ejecución del sistema.

Para que las aplicaciones puedan ser implementadas y ejecutadas, deben desarrollarse a través de diferentes herramientas, las cuales pueden proveer diferentes interfaces de usuario que especificarán la forma de como los componentes pueden ser desarrollados. Estas herramientas están basadas en estándares que permitirán la independencia e interacción entre múltiples plataformas.

A continuación se describen las herramientas utilizadas para el desarrollo del sistema de solicitudes de crédito comercial.

### **JDeveloper**

Es una herramienta IDE que permite diseñar, desplegar y administrar procesos de negocio BPEL, ya que utiliza BPEL como su formato nativo, lo cual hace que los procesos diseñados sean portables hacia distintos servidores de aplicaciones. Además, *JDeveloper* trabaja con lo que se denomina Módulos de Aplicaciones. Este sirve como contenedor de alto nivel para esta aplicación y se compone de un variable número de proyectos. Estos proyectos contienen elementos de trabajo, tales como: clases java, páginas web, ficheros de configuración, etc.

### **Microsoft Visual Basic .Net**

Es un lenguaje orientado a objetos que presenta las cualidades en un entorno integrado de desarrollo, gracias al .Net Framework, que facilita la construcción de aplicaciones Windows. La base .Net Framework permite utilizar cualquier lenguaje perteneciente a Visual Studio .Net, el cual consta de un completo editor de código y más herramientas que facilitan el diseño, programación y depuración de las aplicaciones informáticas que se desarrollarán. Además, utiliza la tecnología ADO.NET como adaptador de base de datos y son llamados objetos dentro de este lenguaje. Estos adaptadores son: *Connection*, *DataAdapter*, *DataSet* que permiten enlazar la interfaz del usuario con un conjunto de datos, mediante consulta de datos con parámetro o procedimientos almacenados.

## PHP

Es una herramienta que se utiliza para construir páginas web dinámicas, ya que su código puede procesarse directamente en el servidor o desde cualquier otro software que permita ejecutar los comandos. En este proyecto se ha utilizado NetBeans 8.0 para ejecutar el código php y diseñar la página web del registro civil. Además, php acepta establecer conexiones a distintas bases de datos, lo cual permite ingresar, modificar, eliminar y consultar datos del contenido de una página web.

### 3.7.2 Diseño de las Interfaces de solicitudes de Crédito

A continuación se describen dos páginas diseñadas, que se construirán con las herramientas de JDeveloper 11g.

En la primera página se van a visualizar todas las solicitudes de crédito registradas, la cual cuenta con un datagrid o lista de columnas de datos registrados, que contendrá unos campos de búsqueda en la parte superior de cada una de las columnas listadas y en la parte superior izquierda, se encontrará una botonera que contendrá un botón para refrescar la página y otro botón para realizar un nueva solicitud de crédito comercial.

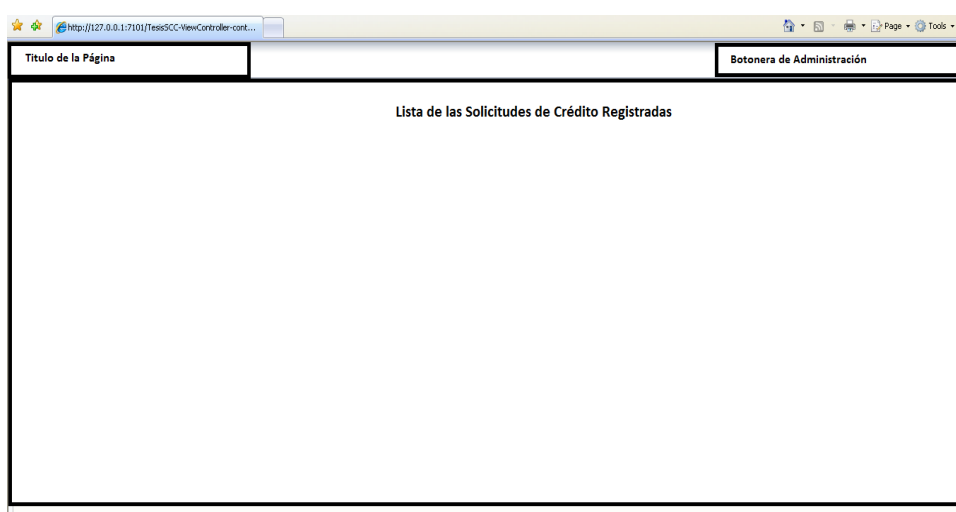


Figura 3.12 Interfaz Lista de Solicitudes de Crédito

En la segunda imagen, se visualiza la página que servirá para ingresar una solicitud de crédito comercial. Consta de varios campos encapsulados en cada una de las pestañas como son: Datos Personales, Información Financiera, Datos del Trabajo y Referencias Personal y Familiar, que servirán para obtener la información necesaria de un cliente que realiza una solicitud de crédito comercial. Además, en la parte superior derecha de esta página, se muestra un estado y el número de solicitud que se mostrará al momento de registrarse. En la parte superior izquierda, se muestra una botonera para administrar la información que se ingrese en la página.

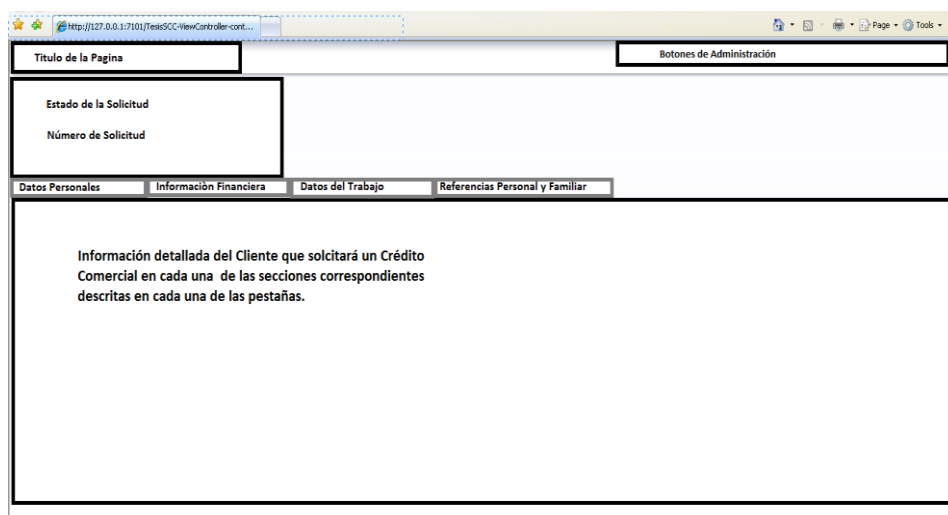


Figura 3.13 Interfaz Ingreso Solicitud de Crédito

### 3.7.3 Diseño de las Interfaces Básicas en PHP y .Net

A continuación se detallan dos páginas diseñadas, que se construirán con las herramientas de PHP y .Net.

En la primera figura se ve la interfaz de la página principal para realizar un registro de propiedad. En esta interfaz se puede encontrar en la parte izquierda un bloque para ingresar los datos personales del propietario y en el bloque de la derecha se debe ingresar los datos de la propiedad. También, se mostraran debajo de los dos bloques una botonera de administración de la página y debajo de esta botonera se visualizará una lista resumen de todas las propiedades que se encuentren registradas.

En la imagen se ve la interfaz gráfica que se construirá en Microsoft Visual Basic .Net.

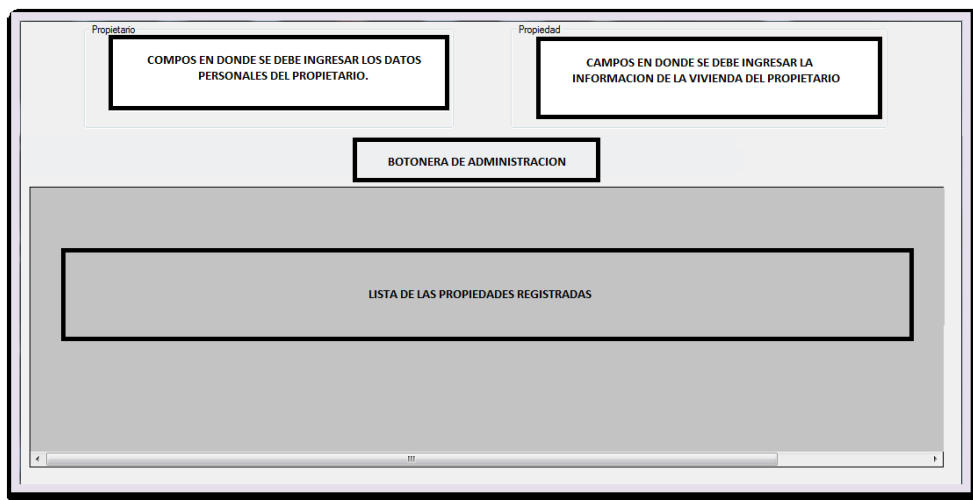


Figura 3.14 Interfaz Registro de la Propiedad

En la segunda figura se encuentra la interfaz de la página principal para realizar el registro civil de una persona. En esta interfaz se puede encontrar el título de la página en la parte superior, debajo de este título se puede ver un bloque para ingresar los datos particulares de una persona. También, más abajo del bloque, estará una botonera de administración de la página y debajo de esta botonera se verá una lista resumen de todas las personas que se encuentren registradas. En la imagen está la interfaz gráfica que se construirá en PHP.

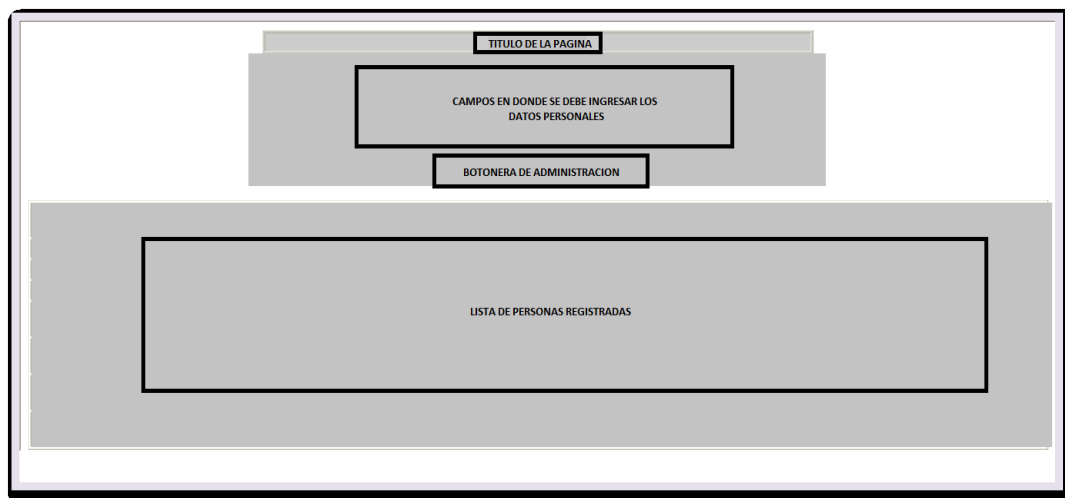


Figura 3.15 Interfaz Registro Civil

### 3.7.4 Diseño de la Base de Datos de solicitudes de Crédito

#### Diseño de la Base de Datos de una Solicitud de Crédito Comercial

A continuación se muestra el diseño Entidad-Relación de la base de datos del Sistema Global de Solicitud de Crédito Comercial.

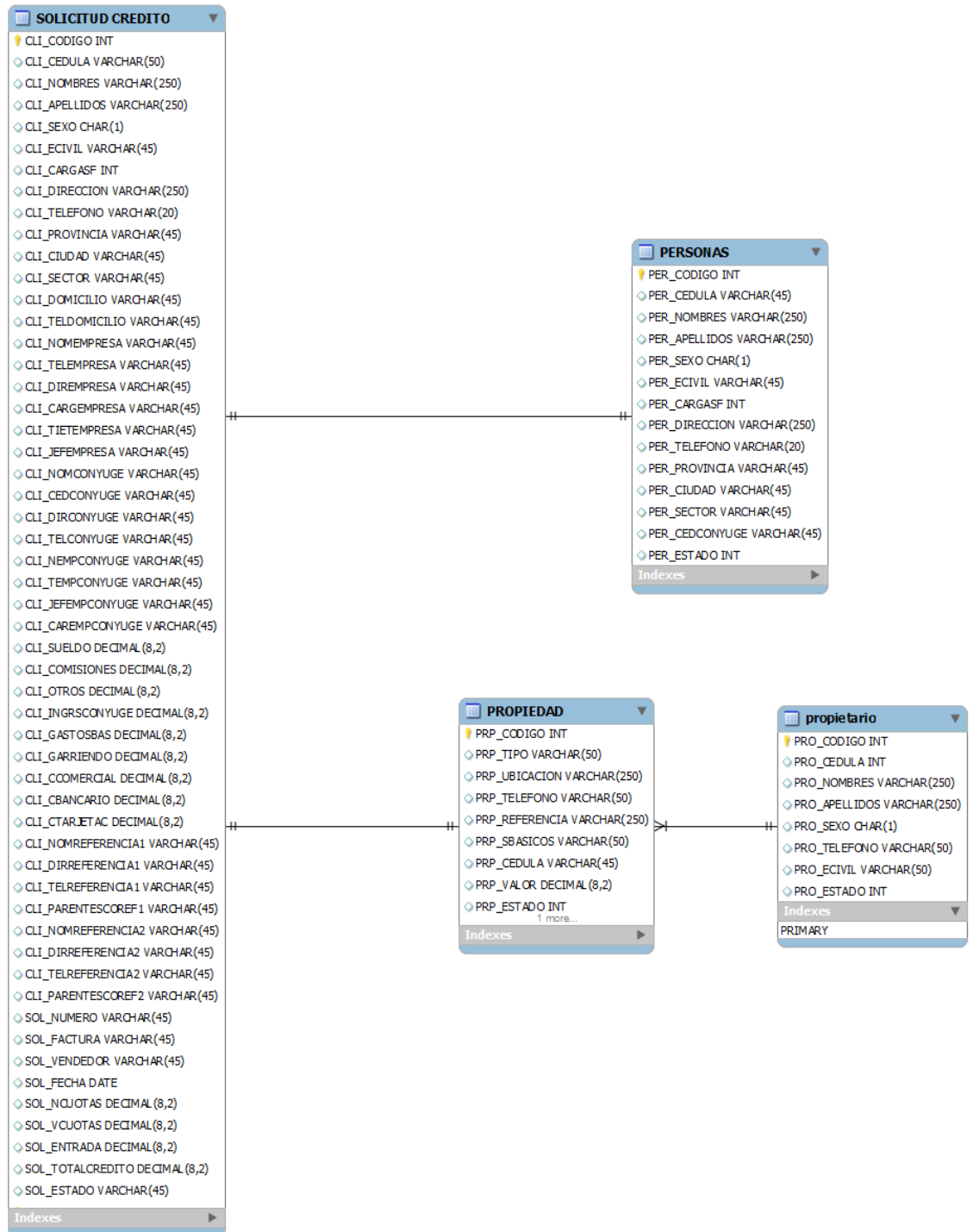


Figura 3.16 Estructura de Tablas de Datos de Solicitud de Crédito

## Diseño de la Base de Datos del Registro de la Propiedad

A continuación se especifica el diseño Entidad-Relación de la base de datos del Registro de la Propiedad.

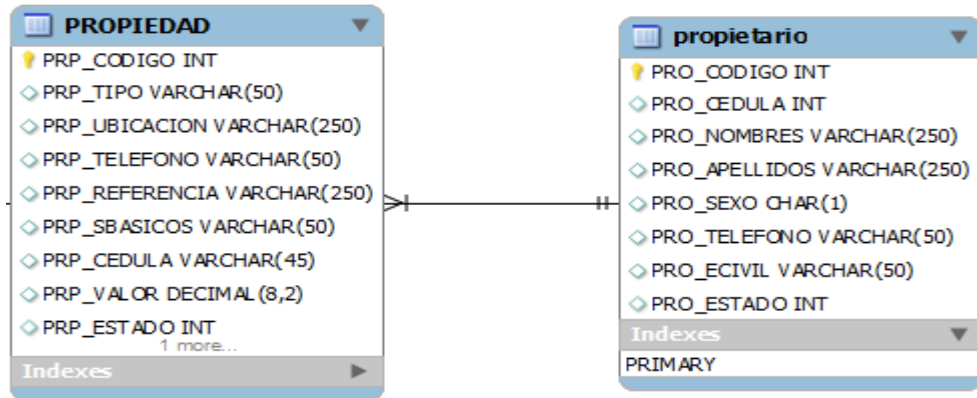


Figura 3.17 Estructura de Tablas de Datos del Registro de la Propiedad

## Diseño de la Base de Datos del Registro Civil

A continuación se ve el diseño Entidad-Relación del Registro Civil.

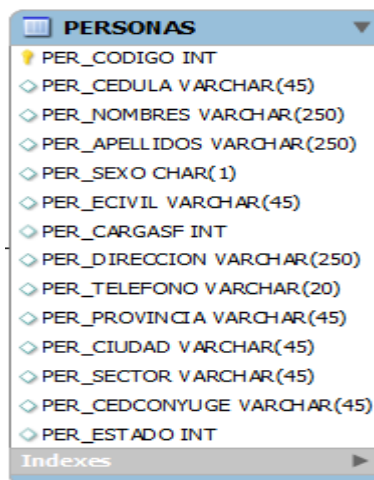


Figura 3.18 Estructura de Tabla de Datos del Registro Civil

Se concluye que en el análisis y diseño de la solicitud de crédito comercial se plasma toda la información recolectada de las entrevistas y de los documentos electrónicos de investigación. Los distintos diagramas de estructuración fueron desarrollados mediante herramientas de Oracle, MySQL y Microsoft para reconocer la interacción de cada compuesto que será utilizado en la construcción de la aplicación de solicitud de crédito.



## CAPITULO 4

### DESARROLLO

En este capítulo se detallan todos los componentes necesarios para el desarrollo de una aplicación de orquestación de servicios. Es decir, se describirán las herramientas de base de datos, orquestación de servicios y de programación, que se utilizarán en los procesos y que culminarán en una aplicación prototipo de solicitudes de crédito comercial.

#### 4.1 Descripción del Ambiente de Desarrollo

El ambiente de desarrollo describe los componentes principales involucrados en este proyecto de tesis, los cuales ayudan a desarrollar, desplegar, ejecutar y demostrar la administración de un flujo de procesos de negocio. Estos componentes se describen a continuación.

**Base de Datos Oracle 11g:** Es la base de datos en donde se van a almacenar las solicitudes de crédito generadas por *BPEL Process Manager*.

**Base de Datos SQL Server 2008:** Es la base de datos en la que se realizarán todas las consultas, de las propiedades simuladas de un registro de la propiedad.

**Base de Datos MySQL 5.5:** Es la base de datos en la que se realizarán todas las consultas, de los datos personales simuladas de un registro civil.

**Oracle BPEL Console 11g:** Es una interfaz de usuario basada en web que se utiliza para administrar, monitorear y manejar los dominios de los procesos e instancias de las Solicitudes de Crédito Comercial.

**Oracle Weblogic Server 11g:** Es una interfaz de usuario basada en web que se utiliza para administrar todas las conexiones de las bases de datos.

**JDeveloper 11g:** Es un IDE muy utilizado para diseñar, compilar, empaquetar y desplegar aplicaciones *BPEL Process* y *ADF Business Component*.

## 4.2 Descripción de las Herramientas para el Desarrollo

Para el desarrollo del sistema de orquestación de servicios, se utilizó Oracle SOA Suite 11gR1 (11.1.1.4.0) y sus componentes necesarios, los cuales se deben descargar de la página de Oracle, y por ende es necesario listar los componentes con su respectivo link. (Hwang, 2011).

### 4.2.1 Definición de las Herramientas para Desarrollo de Orquestación

**WebLogic Server 11g:** Es una herramienta para monitorear las aplicaciones que estén siendo ejecutadas en el sistema de orquestación de servicios y que interactúan con la base de datos.

El Link para descargarse es el siguiente:

<http://download.oracle.com/otn/nt/middleware/11g/wls/wls1034win32.exe>

**RCU (Repository Creation Utility):** Es una herramienta para crear el esquema en la base de datos que serán utilizados en los procesos de la orquestación de servicios.

El Link para descargarse es el siguiente:

[http://download.oracle.com/otn/nt/middleware/11g/111140/ofm\\_rcu\\_win\\_11.1.1.4.0\\_disk1\\_1of1.zip](http://download.oracle.com/otn/nt/middleware/11g/111140/ofm_rcu_win_11.1.1.4.0_disk1_1of1.zip)

**SOA Suite 11.1.1.4.0 (2 partes):** Esta herramienta ayuda a administrar los procesos y servicios web o monitorear las actividades de los negocios que se están ejecutando. También permite que las aplicaciones se conecten con cualquier fuente de datos, creando registros de servicios, que ayudan a administrar el ciclo de vida de los mismos.

Los Links para descargarse las dos partes son los siguientes:

[http://download.oracle.com/otn/nt/middleware/11g/111140/ofm\\_soa\\_generic\\_11.1.1.4.0\\_disk1\\_1of2.zip](http://download.oracle.com/otn/nt/middleware/11g/111140/ofm_soa_generic_11.1.1.4.0_disk1_1of2.zip)

[http://download.oracle.com/otn/nt/middleware/11g/111140/ofm\\_soa\\_generic\\_11.1.1.4.0\\_disk1\\_2of2.zip](http://download.oracle.com/otn/nt/middleware/11g/111140/ofm_soa_generic_11.1.1.4.0_disk1_2of2.zip)

## 4.2.2 Definición de las Herramientas de Bases de Datos

**Database Oracle XE:** Es la base de datos Oracle que se utiliza para guardar la información de los clientes que realizan las solicitudes de crédito.

El Link para descargarse es el siguiente:

<http://download.oracle.com/otn/nt/oracle10g/xe/10201/OracleXEUniv.exe>.

## 4.2.3 Definición de las Herramientas de Programación

**JDeveloper 11.1.1.4.0 :** Es un entorno de desarrollo integrado (IDE), que en la cual ayuda a desarrollar, elaborar y administrar perfiles e implementar los servicios requeridos. (Hwang, 2011)

El Link para descargarse es el siguiente:

<http://download.oracle.com/otn/java/jdeveloper/11.1.1.4.0/jdevstudio11114install.exe>

**Soa-jdev-extension:** Es un paquete de SOA que se importa a *JDeveloper*, por lo que se debe descargar de la página de Oracle o hacer un *update* de esa extensión desde el propio programa, el cual se explica más adelante. (Hwang, 2011)

Los componentes descargados se muestran a continuación:

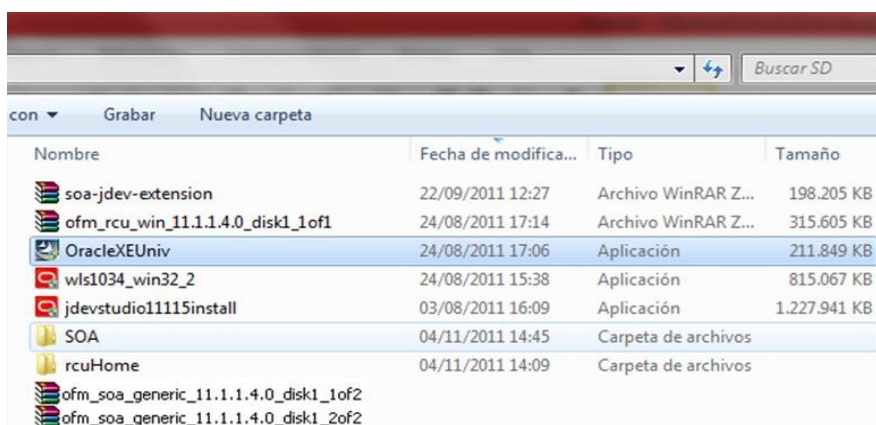


Figura 4.1 Herramientas de Diseño y Programación

Las dos carpetas que vemos serán los descomprimidos de SOA Suite y RCU, el resto son archivos ejecutables.

### 4.3 Configuraciones

A continuación se configuran las herramientas que se utilizarán para el desarrollo del sistema de orquestación de servicios, para lo cual se iniciará primero por la Base de Datos, luego por los componentes de orquestación de servicios y finalmente se concluye con las herramientas de programación.

#### 4.3.1 Configuración de las Herramientas de Bases de Datos

Instalación de Oracle *Database Express Edition*. (Oracle, 2011)

Para instalar la base de datos Oracle XE se ejecuta el archivo **OracleXEUniv.exe**, el cual, en el proceso de instalación solo se tiene que ingresar la contraseña que se utiliza para ingresar como administrador de la base de datos. Por eso, para la instalación en este caso práctico se ingresará con el usuario **system** y la contraseña **miguel**.

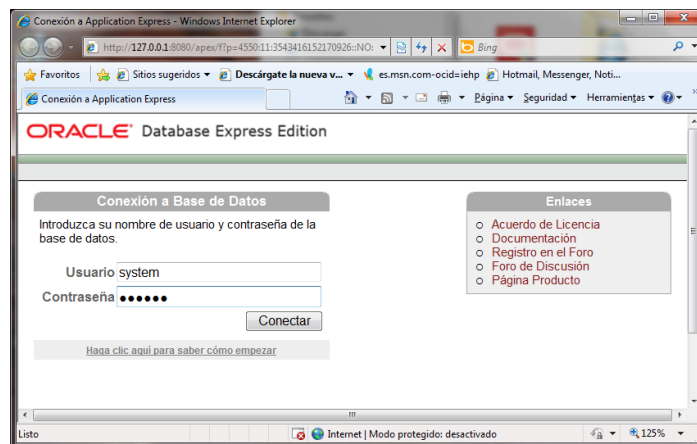
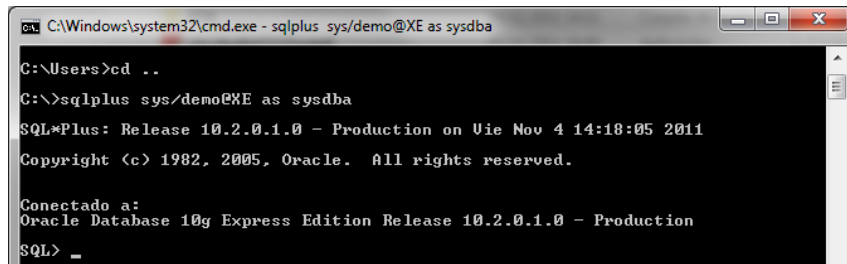


Figura 4.2 Configuración de la Base de Datos

A continuación se configuran los parámetros de la base de datos para que no existan problemas al momento de ejecutar la base de datos XE. Para esto, se utilizará la ventana DOS para ingresar los respectivos comandos, como se muestra en las siguientes figuras:

Primero, se procede a conectar con la base de datos, para esto se debe ingresar con el siguiente comando `c:\sqlplus sys/demo@XE as sysdba`.

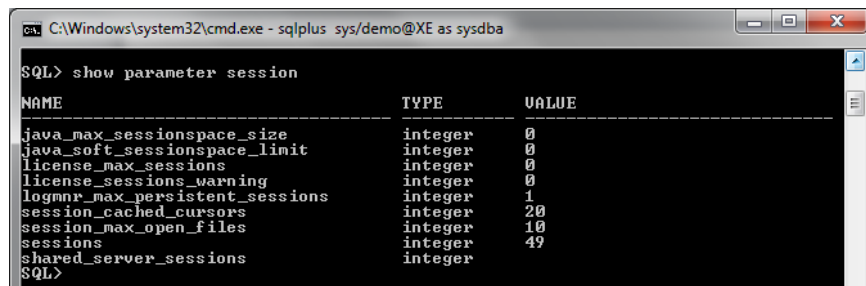


```
C:\Windows\system32\cmd.exe - sqlplus sys/demo@XE as sysdba
C:\Users>cd ..
C:\>sqlplus sys/demo@XE as sysdba
SQL*Plus: Release 10.2.0.1.0 - Production on Uie Nov 4 14:18:05 2011
Copyright (c) 1982, 2005, Oracle. All rights reserved.

Conectado a:
Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production
SQL> _
```

Figura 4.3 Sesión de SQLPLUS

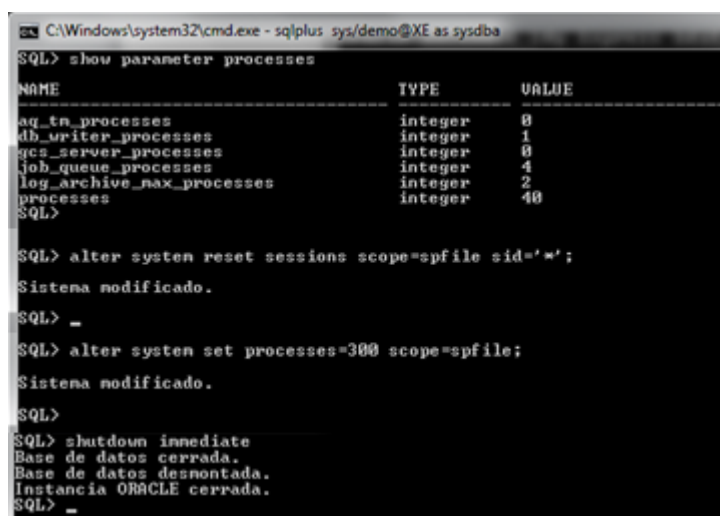
En la siguiente figura están los parámetros a ser modificados. En este caso el parámetro **sessions** es el que va a sufrir las modificaciones.



```
SQL> show parameter session
NAME                                 TYPE        VALUE
-----
java_max_sessionspace_size          integer     0
java_soft_sessionspace_limit        integer     0
license_max_sessions                 integer     0
license_sessions_warning             integer     0
logmnr_max_persistent_sessions      integer     1
session_cached_cursors              integer     20
session_max_open_files               integer     10
sessions                             integer     49
shared_server_sessions              integer
```

Figura 4.4 Parámetros a Modificar

En este otro caso el parámetro **processes** es el que va a sufrir las modificaciones.



```
SQL> show parameter processes
NAME                                 TYPE        VALUE
-----
aq_tm_processes                     integer     0
db_writer_processes                  integer     1
pys_server_processes                 integer     0
job_queue_processes                  integer     4
log_archive_max_processes            integer     2
processes                             integer     40
SQL>
SQL> alter system reset sessions scope=spfile sid='*';
Sistema modificado.
SQL> _
SQL> alter system set processes=300 scope=spfile;
Sistema modificado.
SQL>
SQL> shutdown immediate
Base de datos cerrada.
Base de datos desmontada.
Instancia ORACLE cerrada.
SQL> _
```

Figura 4.5 Parámetros Modificados

Una vez que se haya ingresado todos los comandos, reiniciar el servicio **OracleServiceXE**, para que la base de datos esté lista para almacenar la información.

### 4.3.2 Configuración de las Herramientas de Orquestación

Instalación de *WebLogic Server* 11g. (Hwang, 2011).

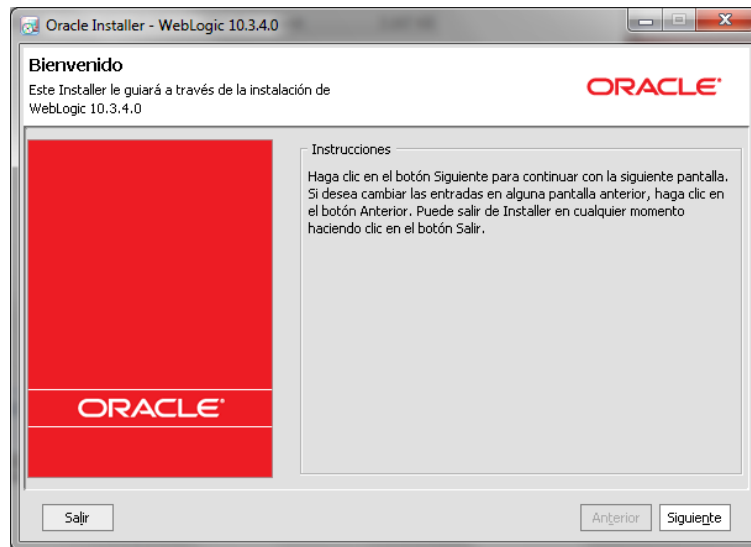


Figura 4.6 Inicio de Instalación

En el siguiente cuadro se elige **Crear un nuevo directorio raíz de Middleware**, luego se ingresa el directorio y se da clic en **siguiente**.

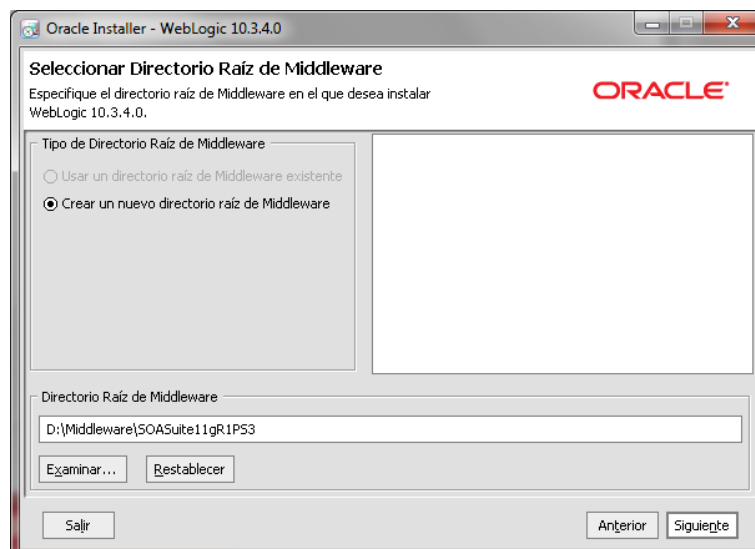


Figura 4.7 Selección del directorio de Instalación

En este caso saltarse el registro para recibir actualizaciones y dar click en **siguiente**.



Figura 4.8 Registro de Seguridad

Escoger **Típica** y dar clic en **siguiente**.

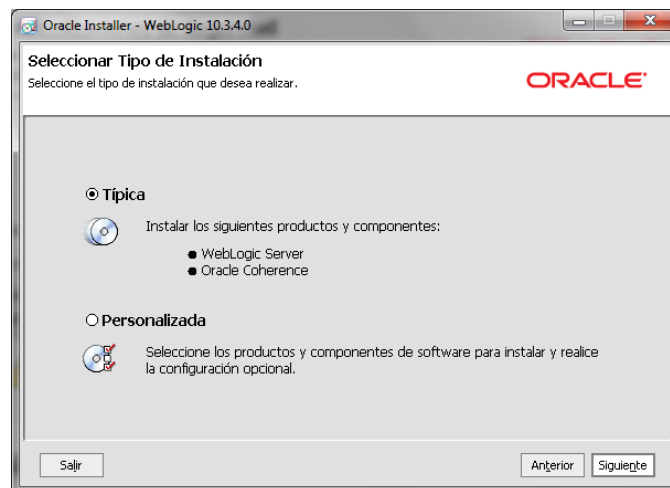


Figura 4.9 Selección de Tipo de Instalación

Clic en **Siguiente**.



Figura 4.10 Selección de Ruta de instalación

Seleccionar **“Todos los Usuarios”** y click en **Siguiente**.

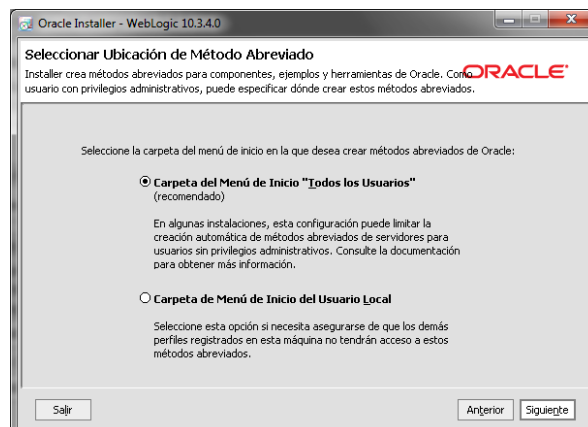


Figura 4.11 Selección de Método Abreviado

Seleccionar la carpeta Menú de Inicio y dar click en **Siguiente**.





Figura 4.12 Resumen de Instalación

Por último, se quita la selección del cuadro **Ejecutar Quickstart** y se da click en **Listo**.

Para la instalación del *schema* de la base de datos se utilizará el instalador *Repository Creation Utility*, por lo que se debe ejecutar el archivo .bat que está ubicado en la carpeta `...\rcuHome\BIN\rcu.bat` descomprimido anteriormente.

**NOTA IMPORTANTE** antes de crear el *schema*.

Si se utiliza Oracle XE como gestor de base de datos, se agregará una nueva variable de entorno en el sistema, para lo cual se procede a realizar lo siguiente:

Ir a propiedades del sistema y seleccionar la pestaña **Opciones Avanzadas** y luego dar clic en el botón **Variables de Entorno**.

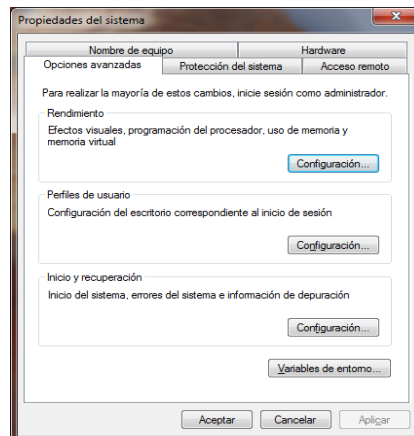


Figura 4.13 Configurar Propiedades del Sistema

Aquí se agregará una nueva variable del sistema y luego se ingresa como nombre **RCU\_JDBC\_TRIM\_BLOCKS** con el valor de **true** y se pulsa **aceptar**.

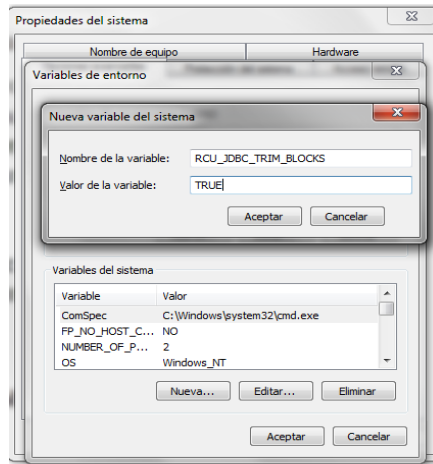


Figura 4.14 Inserción de Variable del Sistema

Después se procede a instalar el *schema*, dando click en **rcu.bat**, como se puede ver en el siguiente recuadro.

Dar clic en **Siguiente**.

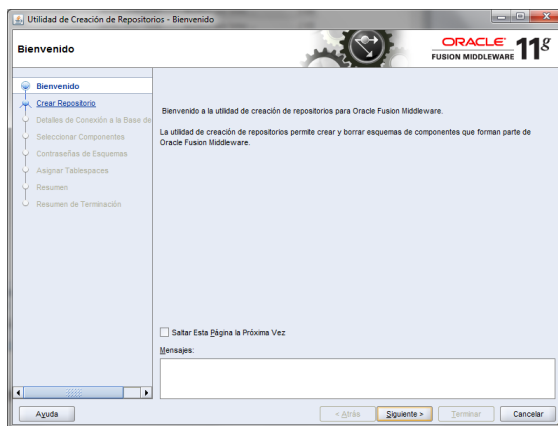


Figura 4.15 Cuadro de Bienvenida de RCU

Seleccionar **Crear** y dar clic en **Siguiente**.



Figura 4.16 Creación o Eliminación de Repositorio

En el siguiente recuadro se ingresarán los siguientes datos y dar click en **Siguiente**.



Figura 4.17 Conexión a la base de datos

Se ignora el siguiente mensaje que salga, dar click en **aceptar** y luego en **Siguiente**.

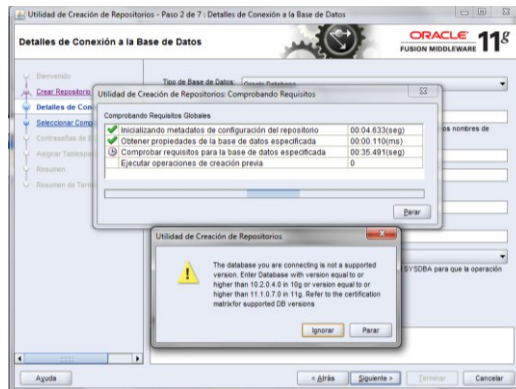


Figura 4.18 Estado de Configuración de Conexión

En la columna de componentes se escoge **Infraestructura SOA y BPM** y click en **Siguiente**.



Figura 4.19 Estructura de Selección de Componentes

Dar click en **aceptar** y luego en **Siguiente**. Luego se debe establecer la contraseña **demo** para todos los esquemas.



Figura 4.20 Contraseñas de Esquema

Click en **aceptar** y luego en **siguiente**.

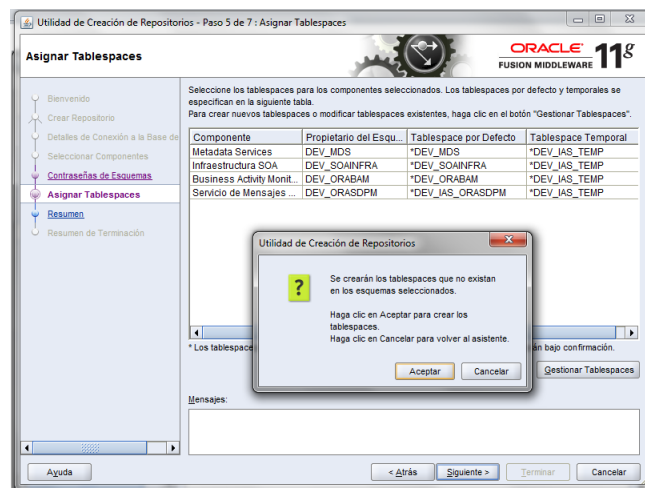


Figura 4.21 Asignación de Tablespace

Click en **crear** y se espera a que se creen los esquemas para pulsar **cerrar**.



Figura 4.22 Detalles de Terminación

## Instalación de SOA Suite

Una vez que hayan sido creados los *schemas*, se puede instalar el SOA *server* ejecutando SOA\Disk1\setup.exe. Luego hay que localizar la dirección del directorio JDK, llamado jdk160\_21 dentro de la carpeta Middleware, y después se debe ingresar en la ventana que se muestre:

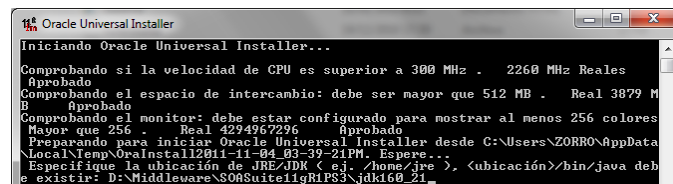


Figura 4.23 Ingreso de ubicación del JDK

Luego de dar click en siguiente, omitir el registro de actualizaciones y dar click en siguiente hasta que su instalación sea finalizada.



Figura 4.24 Comprobación de Requisitos



Figura 4.25 Ubicación de Instalación



Figura 4.26 Selección del Servidor de Aplicaciones

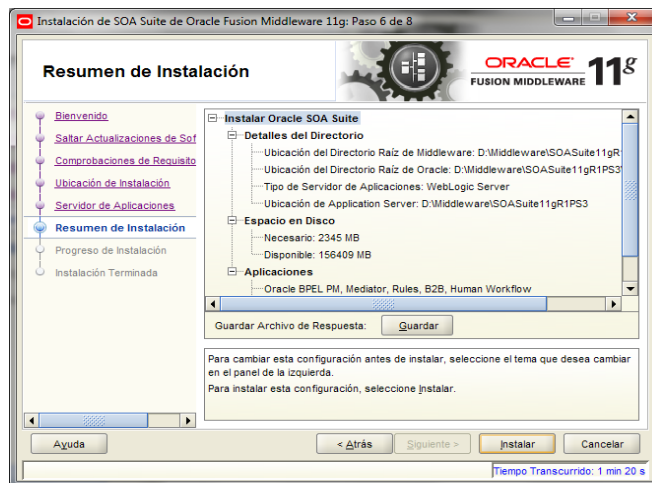


Figura 4.27 Detalle de Instalación

Finalizada la instalación dar click en terminar y la instalación estará completa.

### Crear un dominio de SOA

Para lo cual hay que ir a Inicio->Todos los programas->Oracle Common Home 11g - Home1 ---->Configure Application Server. (Oracle, 2011)

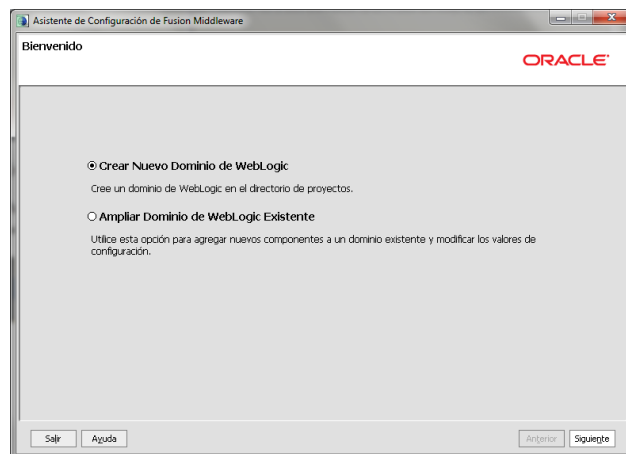


Figura 4.28 Creación del Dominio SOA

Seleccionar los productos para generar un dominio.

Oracle SOA Suite for developers – 11.1.1.0 [Oracle\_SOA1]:  
Seleccionándolo, se seleccionaran automáticamente Oracle WSM Policy  
Manager and Oracle JRF

Oracle Enterprise Manager – 11.1.1.0 [oracle\_common]

Oracle Business Activity Monitoring – 11.1.1.0 [Oracle\_SOA1] if you want  
to use BAM

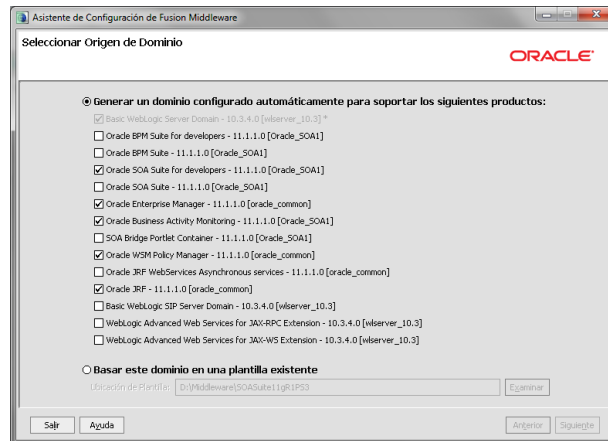


Figura 4.29 Selección de Dominios



Figura 4.30 Especificación del Nombre y Ubicación del Dominio

A continuación, se debe ingresar el usuario **weblogic** y la contraseña  
**demooracle1**.



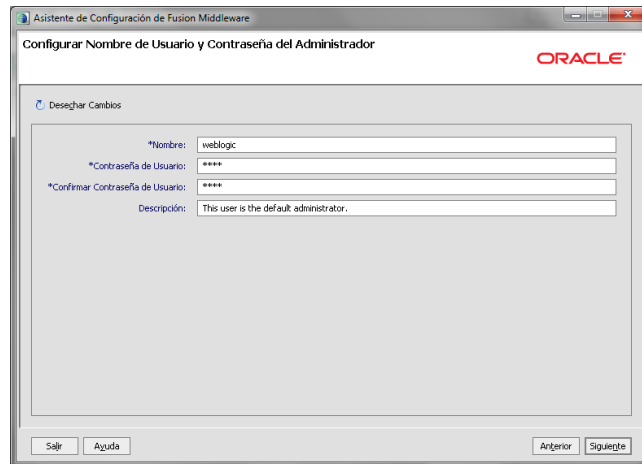


Figura 4.31 Ingreso de Usuario y Contraseña del dominio

La clave **demo** se ingresa para todos los esquemas y también se debe llenar todos los datos correctos en los campos correspondientes.

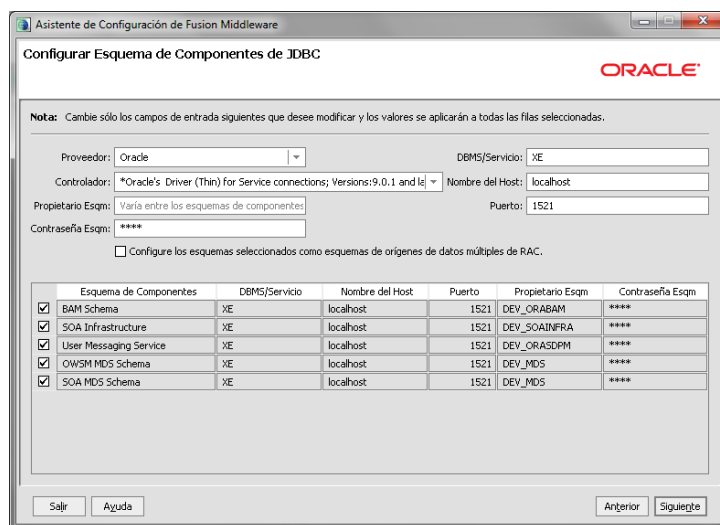


Figura 4.32 Configuración de esquema de componentes JDBC

Dar click en siguiente, luego en finalizar y el servidor está instalado.

### 4.3.3 Configuración de las Herramientas de Programación

#### Instalación de JDeveloper 11g. (Oracle, 2011).

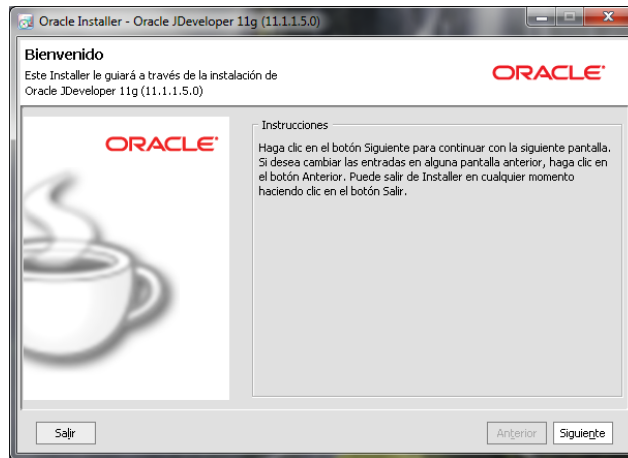


Figura 4.33 Instalación Oracle JDeveloper

A continuación se selecciona la ubicación donde debe instalarse la herramienta de Oracle JDeveloper 11g.

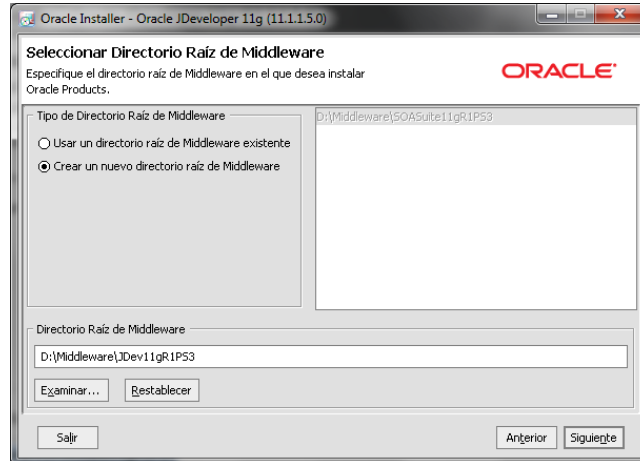


Figura 4.34 Selección del directorio raíz de Middleware

Después, se da click en siguiente y se espera una siguiente pantalla para quitar la selección de ejecutar Quickstart y dar click en Listo.



Figura 4.35 Finalización de Instalación

A continuación se detallará la implementación de una herramienta para interactuar con SOA. Primero se debe seleccionar inicio, luego se busca la carpeta Oracle Fusion Middleware 11.1.1.5.0 y se da click en la herramienta JDeveloper Studio 11.1.1.5.0 o de cualquier versión.

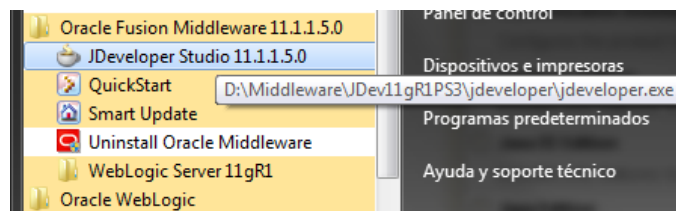


Figura 4.36 Ejecución de JDeveloper Studio 11.1.1.5.0

Seleccionar el rol por default que se requiere y dar click en OK.

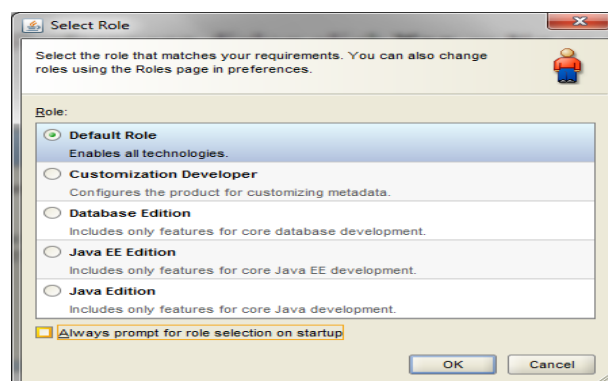


Figura 4.37 Selección del Rol

Actualizar JDeveloper agregando el pack de soa-jdev-extension, para lo cual, hay que ubicarse en la barra de menú y se debe seleccionar Help->CheckforUpdates.

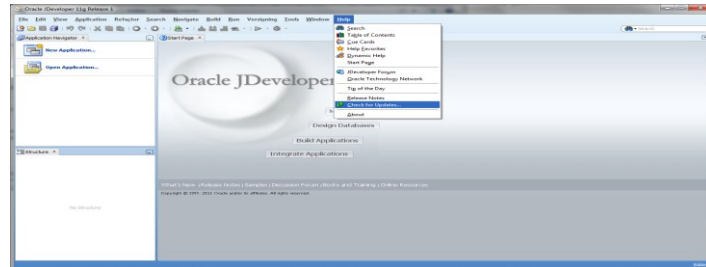


Figura 4.38 Selección del menú de actualización

Dar click en siguiente para ver la opción de actualizar JDeveloper mediante internet o por medio de una ubicación específica del archivo.

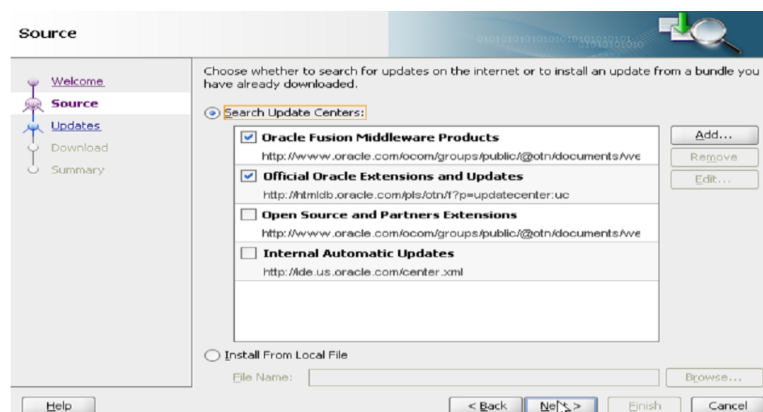


Figura 4.39 Selección de tipo de actualización

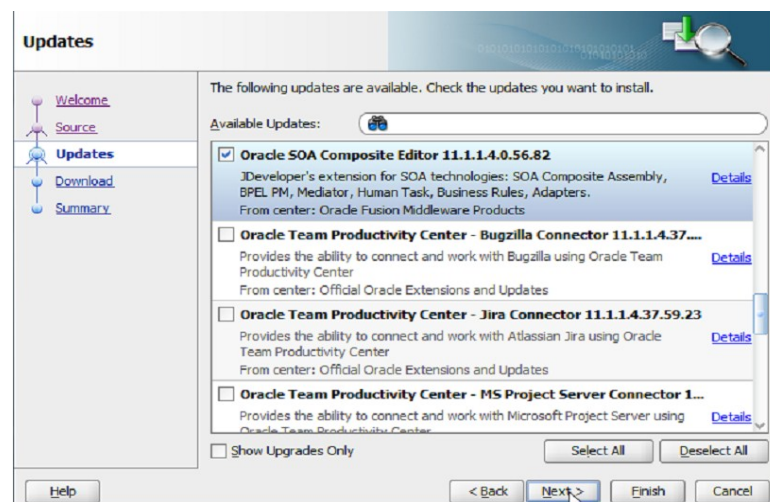


Figura 4.40 Selección de componente de actualización

El componente de SOA esta agregado, se da un click en *finish* y saldrá un mensaje en el que se debe dar click en *yes* para que se reinicie JDeveloper y se actualicen sus componentes.

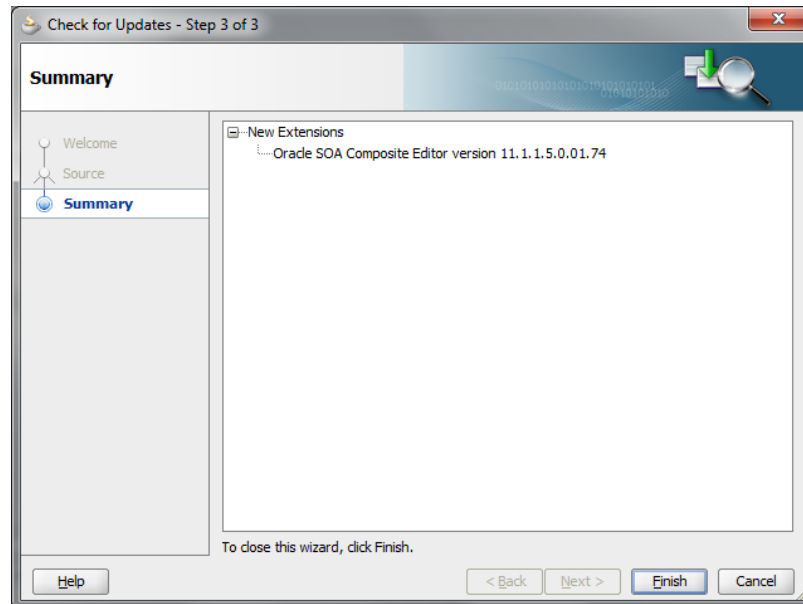


Figura 4.41 Resumen de Actualización SOA

A continuación se determina los límites de memoria que se necesitan para que el rendimiento del sistema sea el más óptimo.

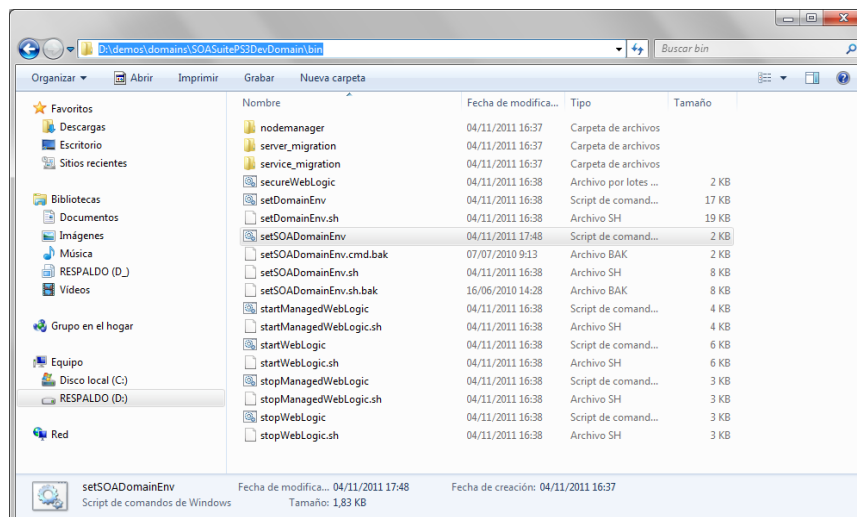


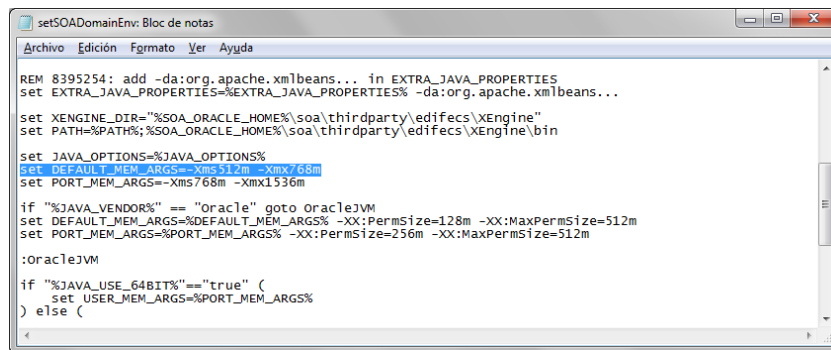
Figura 4.42 Selección del archivo de configuración

Se edita el archivo `setSOADomainEnv.cmd` y se cambia.

```
set DEFAULT_MEM_ARGS=-Xms512m -Xmx1024m
```

por

set DEFAULT\_MEM\_ARGS=-Xms512m -Xmx768m .



```
setSOADomainEnv: Bloc de notas
Archivo Edición Formato Ver Ayuda
REM 8395254: add -da:org.apache.xmlbeans... in EXTRA_JAVA_PROPERTIES
set EXTRA_JAVA_PROPERTIES=%EXTRA_JAVA_PROPERTIES% -da:org.apache.xmlbeans...
set XENGINE_DIR="%SOA_ORACLE_HOME%\soa\thirdparty\edifecs\XEngine"
set PATH=%PATH%;%SOA_ORACLE_HOME%\soa\thirdparty\edifecs\XEngine\bin
set JAVA_OPTIONS=%JAVA_OPTIONS%
set DEFAULT_MEM_ARGS=-Xms512m -Xmx768m
set PORT_MEM_ARGS=-Xms768m -Xmx1536m
if "%JAVA_VENDOR%" == "Oracle" goto OracleJVM
set DEFAULT_MEM_ARGS=%DEFAULT_MEM_ARGS% -XX:PermSize=128m -XX:MaxPermSize=512m
set PORT_MEM_ARGS=%PORT_MEM_ARGS% -XX:PermSize=256m -XX:MaxPermSize=512m
:OracleJVM
if "%JAVA_USE_64BIT%"=="true" (
  set USER_MEM_ARGS=%PORT_MEM_ARGS%
) else (
```

Figura 4.43 Configuración de argumentos

Se debe ejecutar el archivo startWebLogic.cmd para levantar los servicios.

### Ingreso a las consolas

Para ingresar al administrador de weblogic server, se ingresa con el usuario **weblogic** y con la contraseña **demooracle1** que se ingresó con anterioridad para instalar el servidor weblogic.

Weblogic console: <http://localhost:7001/console>

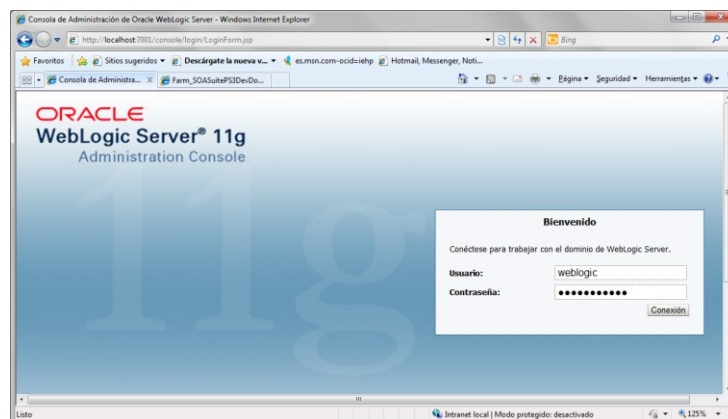


Figura 4.44 Inicio de sesión a la consola Weblogic Server

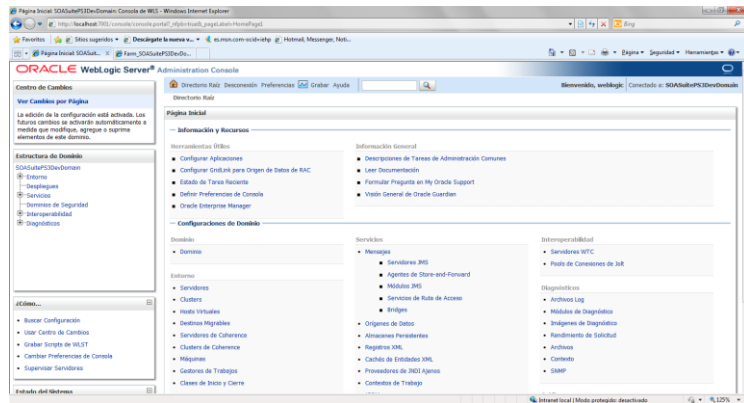


Figura 4.45 Configuración de la consola de Weblogic Server

Dar click en **despliegue**.



Figura 4.46 Selección del dominio de configuración

Después, se hace click en personalizar esta tabla y luego se selecciona la opción de excluir bibliotecas al mostrar despliegues. También, se modifica el número de filas por página a 100, ya que es necesario cuando existe una consulta de datos de alrededor de 100 registros. Esto dependerá de la capacidad de información que se emplee en una organización. Luego dar click en aplicar.

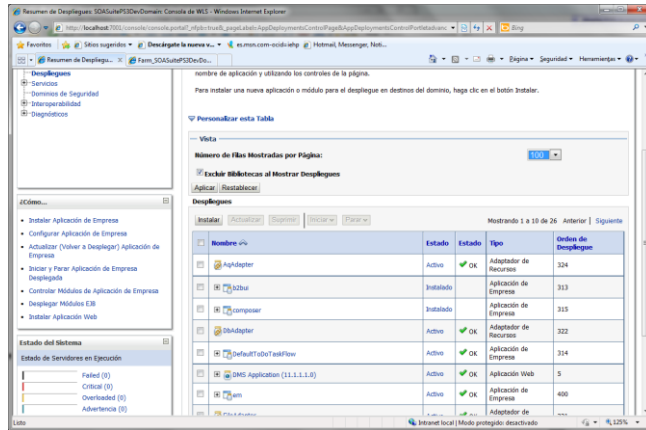


Figura 4.47 Personalizar Tabla

Se da ahora en el nombre del dominio que se encuentra en la parte superior del menú. Luego se selecciona la pestaña Configuración y se quita la opción **Activar despliegue a petición de aplicaciones internas** y después click en el botón guardar.

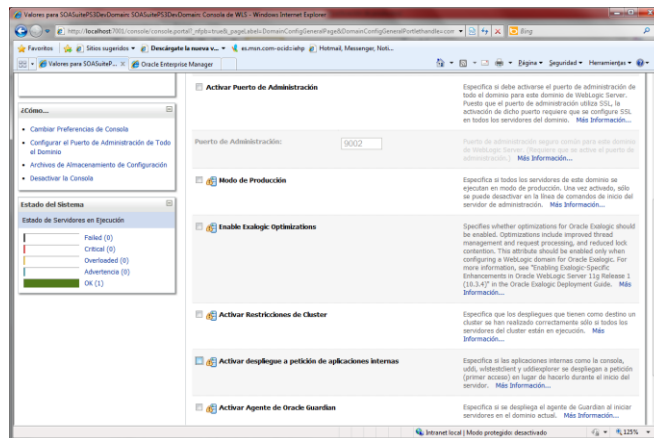


Figura 4.48 Configuración de Despliegue

De la misma forma, se debe iniciar con el usuario **weblogic** y con la contraseña **demooracle1** para ingresar a la consola de Administración Empresarial.

Enterprise Manager console: <http://localhost:7001/em>.



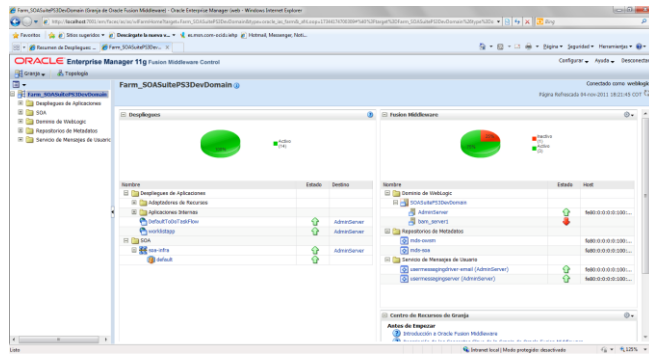


Figura 4.49 Consola de Administración Empresarial

En el menú que se encuentra a la izquierda de la página, se debe desplegar la carpeta SOA y luego dar click derecho en soa-infra(AdminServer) e ir a Administración SOA → Propiedades Comunes.

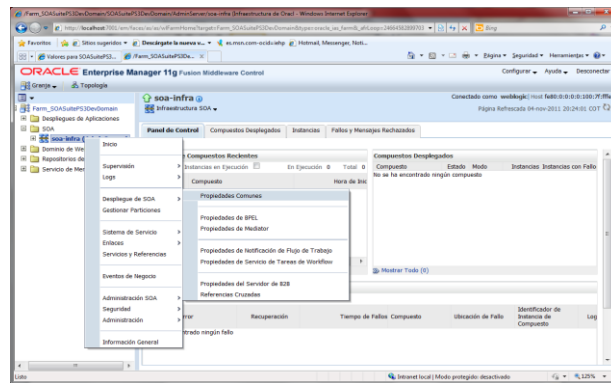


Figura 4.50 Selección de Propiedades Comunes

A continuación en **Nivel de Auditoria** se escoge Desarrollo y en **Captura Estado de Instancia de Compuerto** dar click en el checkbox, después se da click en el botón **aplicar** y click en **sí**.

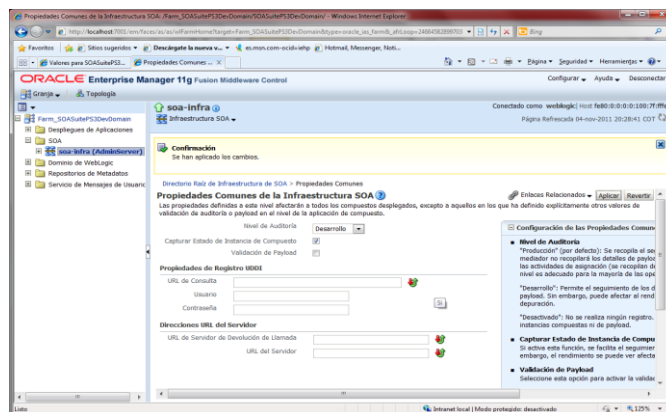


Figura 4.51 Selección de nivel de Auditora

## 4.4 Creación de las Interfaces Gráficas del Sistema

### Solicitud de Crédito Comercial

Para el diseño de la página principal del sistema, inserción de texto, botones e información externa en el sistema, se debe usar la herramienta JDeveloper que permite el diseño de interfaces web mediante *ADF Business Component* al igual que su programación. (Walter Egan, 2011)

A continuación se presenta la interfaz de trabajo de JDeveloper.

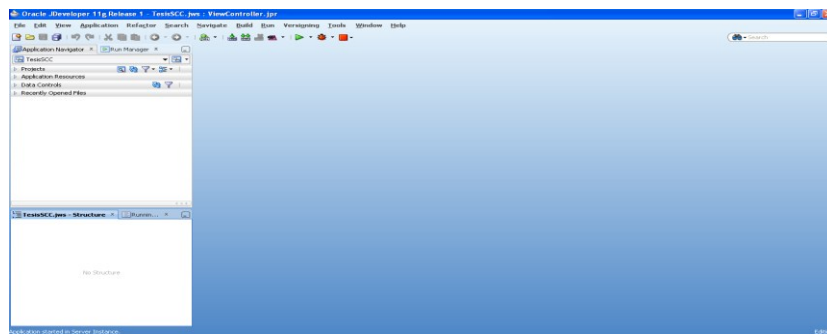


Figura 4.52 IDE Oracle JDeveloper 11g

El sistema está diseñado en dos páginas de extensión .jspx que son páginas que pueden visualizar fácilmente componentes con extensiones gráficas y tabulares desplegando y analizando los datos contenidos, lo cual permite interactuar dinámicamente con la información. (Walter Egan, 2011).

La siguiente página que será la principal, consta de 5 pestañas que contienen los campos necesarios para ingresar la información que se requiere en una solicitud de crédito comercial. También, podrá realizar una nueva solicitud, grabar las nuevas solicitudes e ir a la página de registro de solicitudes. El botón buscar servirá para encontrar los clientes registrados en las solicitudes.

La primera pestaña “Datos Personales”, contendrá toda la información personal del cliente. Además, contiene un campo especial que servirá para que el vendedor ingrese la cédula del cliente y consulte sus datos personales, ya sea desde la propia base de la empresa que registra las solicitudes de crédito o desde la base externa de un registro civil.

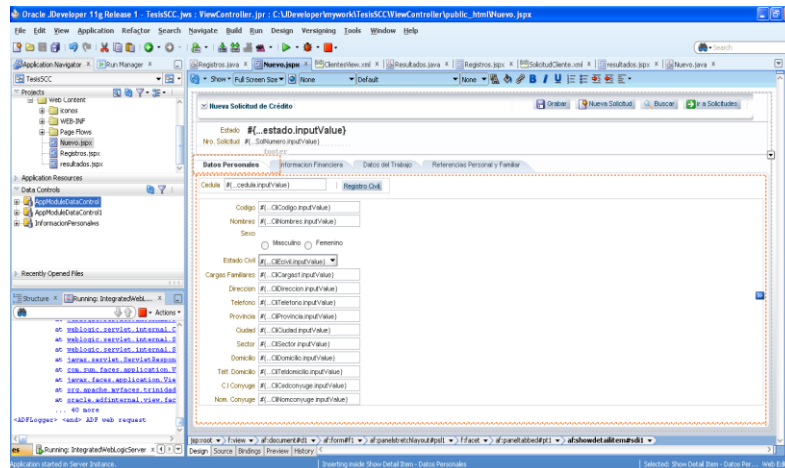


Figura 4.52 Detalles de la Pestaña Principal

La segunda pestaña “Información Financiera”, contendrá toda la información de los ingresos y egresos del cliente. Una vez obtenida esta información el vendedor deberá ingresar la información del crédito comercial solicitado, para después ejecutar los procesos de toma de decisión automática por medio del botón **Calcular**.

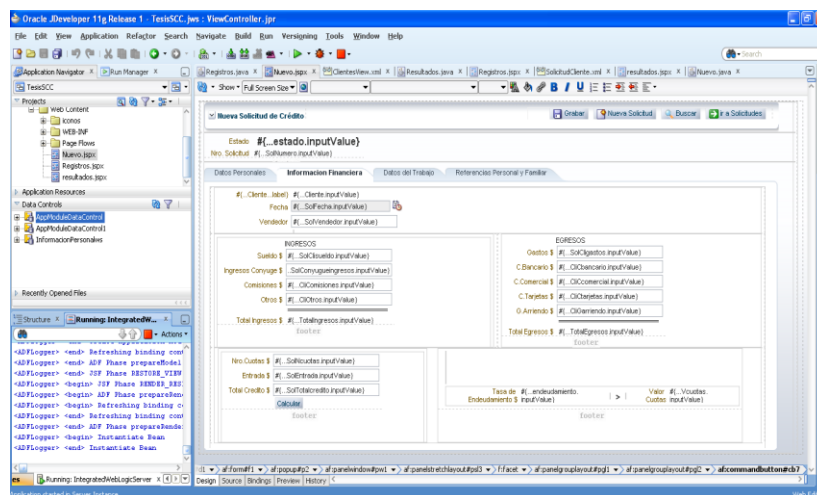


Figura 4.53 Pestaña de Información Financiera

La tercera pestaña “Datos del Trabajo”, contendrá toda la información referente a las empresas en donde trabajan, tanto el cliente que solicita el crédito como su cónyuge, en el caso de ser casado, de lo contrario se dejaría en blanco estos campos de trabajo del cónyuge.

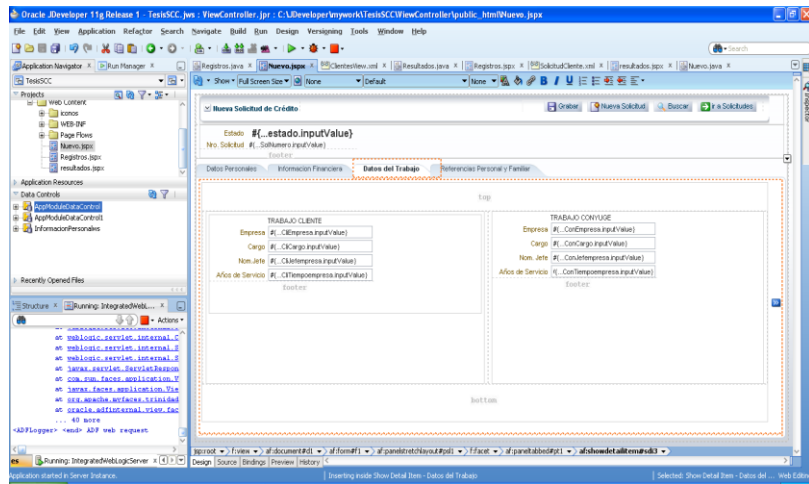


Figura 4.54 Pestaña de Referencias Datos del Trabajo

La cuarta pestaña “Referencias Personal y Familiar”, contendrá toda la información de familiares o amigos que el cliente crea que lo conocen o tengan un vínculo parentesco con el cliente.

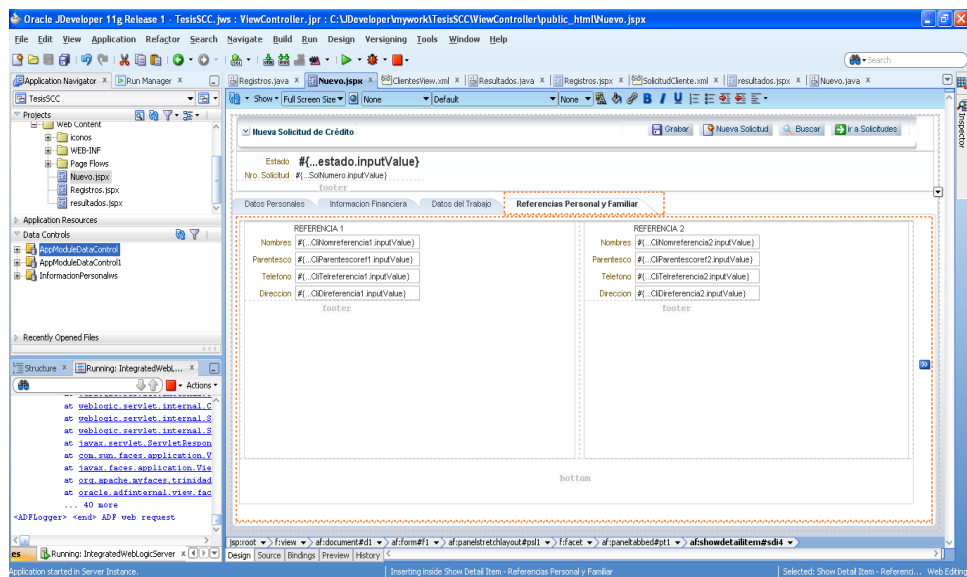


Figura 4.55 Pestaña de Referencias Personales

En la segunda página se visualizarán todos los registros de las solicitudes grabados en la base de datos Oracle. También, el vendedor podrá realizar las consultas de los clientes, ingresando varios datos claves que pertenezcan al cliente en los campos de consulta correspondientes.

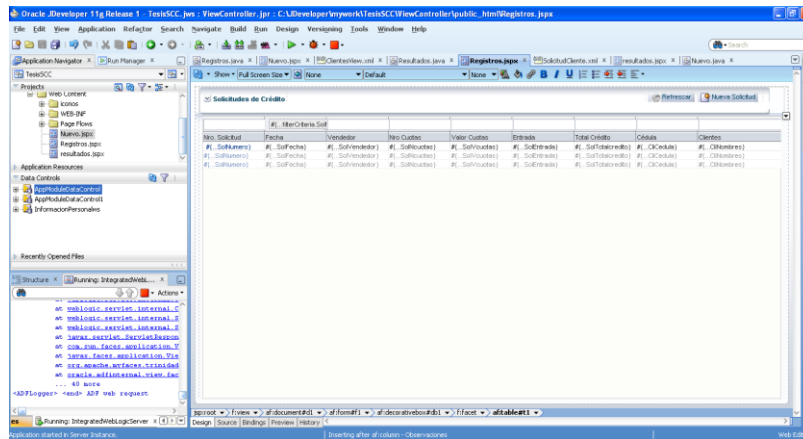


Figura 4.56 Ventana de Consulta de Clientes

La tercera página se encarga de mostrar detalladamente la información que se ha registrado en la página principal y además consta de una botonera que contiene un botón para ir a la página de las solicitudes registradas, un botón para realizar una nueva solicitud y un botón para tomar decisiones de estados de una solicitud de crédito comercial. Los estados de una solicitud de crédito son:

**Aprobado:** Con este estado el cliente podrá acceder directamente al crédito comercial solicitado.

**Pendiente:** Este estado quiere decir que la información obtenida del cliente es insuficiente para obtener un crédito comercial, por lo que el gerente de ventas está a la espera del cambio de una información más actualizada, para tomar la decisión de aprobar o no la solicitud de crédito comercial.

**Rechazado:** Con este estado se niega toda posibilidad de poder tener un crédito comercial.

El tipo de decisión que se tome puede ser de forma manual o automática, ya que el sistema está diseñado para que mediante un cálculo, basado en los ingresos, egresos y tasa de endeudamiento que el cliente tenga, pueda tomar un tipo de decisión. Este tipo de decisiones se describen a continuación.

**Manual:** Este tipo de decisión se ejecuta cuando la solicitud tiene un estado **Pendiente** y el gerente quiere **Aprobar** o **Rechazar** la solicitud requerida por un cliente. Igualmente los demás estados pueden ser escogidos y aceptados dependiendo de cada política de empresa.

**Automática:** Esta decisión en cambio es procesada en el momento en que se obtiene la información financiera del cliente, ya sea para que la solicitud este aprobada, rechazada o pendiente.

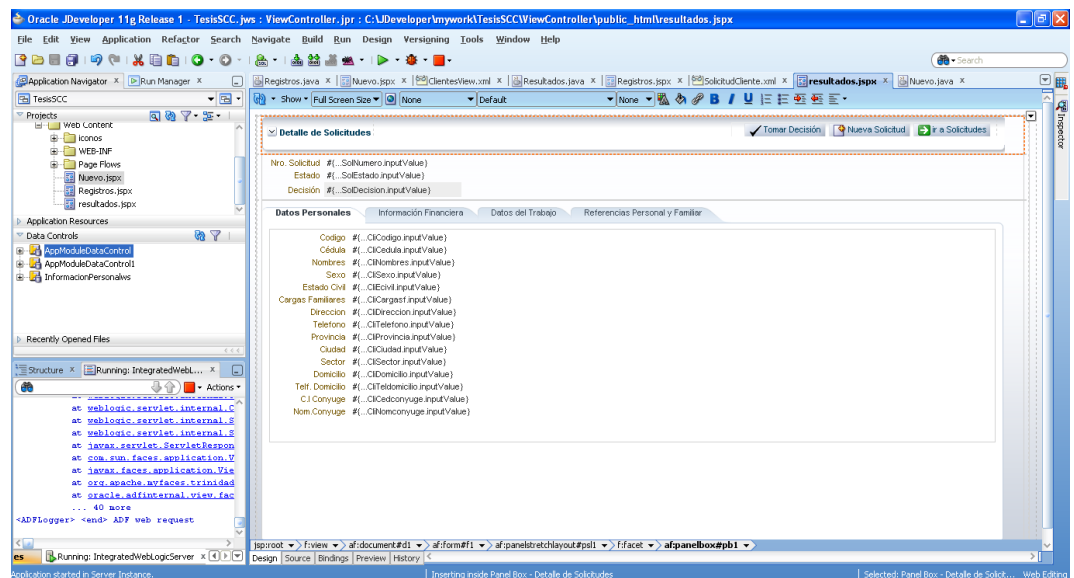


Figura 4.57 Información detallada del Cliente

## Registro Civil

Para el diseño de la página principal del sistema de Registro Civil, inserción de texto y botones, se usará la herramienta NetBeans 8.0 que permite trabajar con código PHP para realizar el diseño de páginas web dinámicas.

La página creada se puede ver a continuación. Esta página permitirá al usuario guardar, actualizar, eliminar y consultar a las personas que estén registradas en la base de datos.

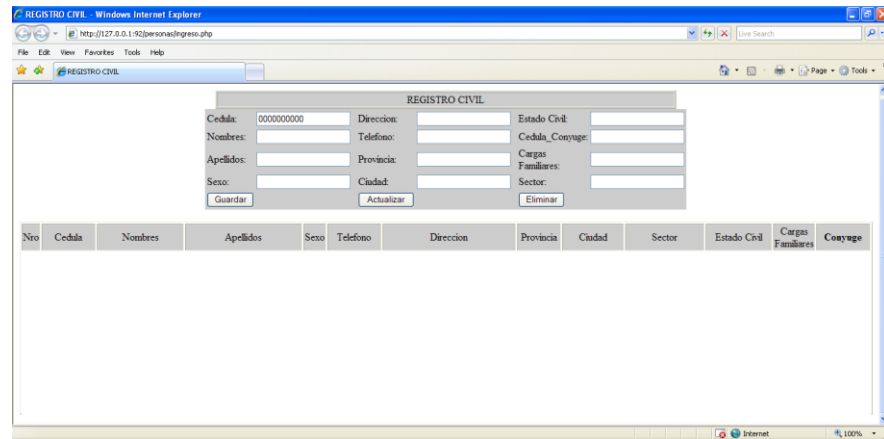


Figura 4.58 Página de Simulación de un Registro Civil

### Registro de la Propiedad

Para el diseño de la página principal del sistema de Registro de la Propiedad, inserción de texto y botones, se usara la herramienta Microsoft Visual .Net que permite el diseño de interfaces mediante el Framework de .Net al igual que su programación. (Microsoft, 2012)

La interfaz creada se puede visualizar a continuación. Esta interfaz permitirá al usuario guardar, actualizar, eliminar y consultar a las personas que tengan registradas sus propiedades en la base de datos.

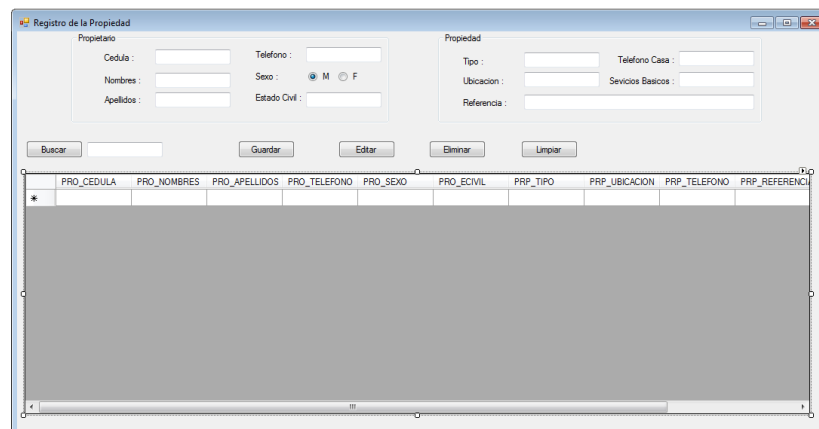


Figura 4.59 Página de Simulación de un Registro de Propiedad

## 4.5 Procedimientos para el desarrollo del Sistema de Solicitud de Crédito

### 4.5.1 Procedimientos Preliminares

Creación de Conexiones: Para enlazar las bases de datos con la interfaz del usuario, es necesario definir primero las conexiones. Estas conexiones se las debe hacer desde el servidor Oracle WebLogic Server, que permite crear orígenes de datos de varias bases de datos. Estos orígenes de datos o JDBC, no ayudaran a sincronizar las bases de datos locales o remotas. **(Beecher, 2011)**

A continuación se detalla el proceso que se debe hacer para crear los JDBC.

1. Primero se debe ingresar a la consola de administración digitando el nombre del usuario(weblogic) y el password(welcome1).
2. Una vez que se ingrese a la consola de administración, hay que dirigirse a la sección **Estructura de Dominio** y luego desplegar la opción **Servicios**, para después seleccionar **Orígenes de Datos**. Una vez que se haya seleccionado la opción hay que ir a la parte derecha y dar click en **Nuevo** para que se despliegue una lista de tipos de orígenes de datos y entonces se escogerá **Origen de Datos Genérico**.
3. Luego de seleccionar el nuevo origen de datos genérico, se visualizará una página para la creación del JDBC. En esta página se debe ingresar el nombre del origen de datos como por ejemplo **OracleConexion**; después, se debe ingresar el JNDI, que para este caso se debe poner **“jdbc/OracleConexion”**, que va a ser la ruta en el servidor WebLogic. Por último, se seleccionará el tipo de base de datos que puede ser Oracle, MySQL o SQLServer y después dar click en siguiente.
4. En la segunda ventana se va a seleccionar el driver de la base de datos que se ha escogido, para realizar diferentes tipos de operaciones. En este caso se ha seleccionado **\*Oracle driver for Service Connections versión 9.0.1 and later** y luego se da click en siguiente.



5. En la tercera ventana el asistente pregunta si desea utilizar transacciones globales y se da click en el checkbox si va a tenerlas o caso contrario se quita la selección y se da click en siguiente.

6. En la cuarta ventana el asistente pide que se llene las propiedades de la conexión de la base de datos, por lo que se debe llenar estos datos correctamente o caso contrario tendrá complicaciones para crear el origen de datos. Luego click en siguiente.

7. En esta ventana se comprueba la conexión a la base de datos. El éxito de la conexión correcta dependerá de la configuración que se haya hecho en la página anterior. Si la conexión es correcta dar click en siguiente.

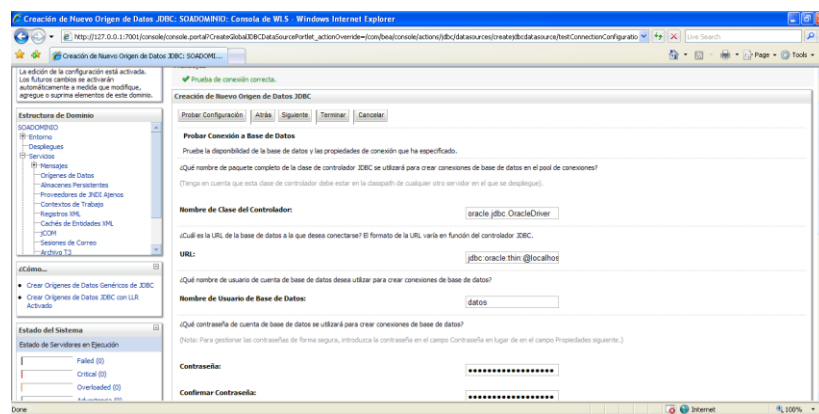


Figura 4.60 Configuración de Conexión con la base de datos

8. Por último, si todo está bien, se selecciona el servidor en el que se quiere que se despliegue el origen de datos. Para este caso se ha seleccionado AdminServer y luego dar click en terminar.

Una vez que se ha creado el origen de datos, se escoge **Estructura de Dominio** y luego se elige **Despliegues**, ahí se muestra un listado de todas las aplicaciones desplegadas en el servidor weblogic, por lo que se debe seleccionar el despliegue DBAdapter para crear un nuevo pool de conexión saliente.

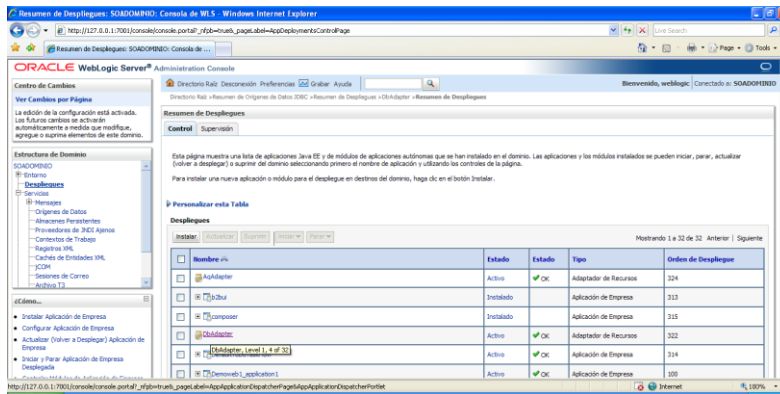


Figura 4.61 Estructura de Dominio

Luego de haber seleccionado el despliegue **DBAdapter** dar click en la pestaña **Configuración** y luego en **Pool de Conexiones Salientes**, para después proceder a crear una nueva conexión saliente, por lo que se debe dar click en el botón **Nuevo**.

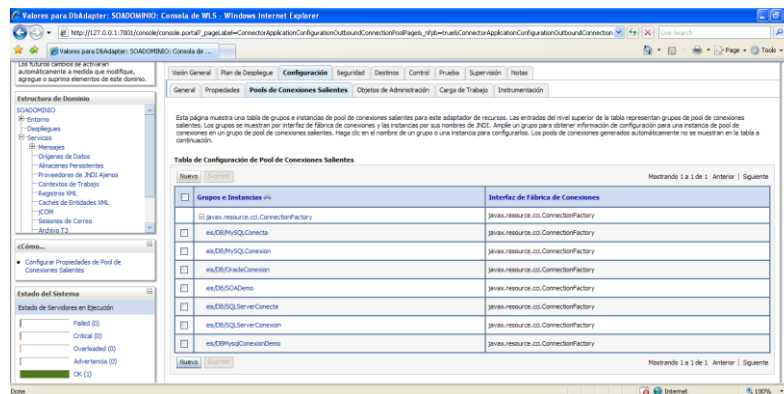


Figura 4.62 Configuración del Pool de Conexiones Salientes

Después de haber hecho click en **Nuevo** aparece una ventana en que se debe escoger el grupo de conexiones saliente a la que desea crear una instancia de esa conexión y luego se debe seleccionar **java.resource.cci.ConnectionFactory**.

Para continuar con la configuración, se debe dar click en siguiente y se mostrara una nueva página, en la cual hay que ingresar el nombre del JNDI para enlazar las conexiones desde el JDeveloper, cada vez que se realice una conexión a la base de datos Oracle. El nombre o ruta que se ingresa es **es/DB/OracleConecta**, la cual servirá para identificar la conexión creada en este punto.

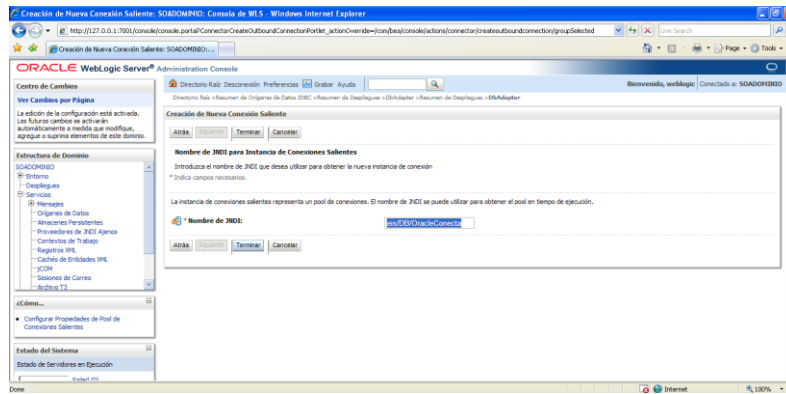


Figura 4.63 Ingreso de la ruta del JNDI

En la siguiente gráfica se ve la conexión saliente creada en el pool de conexiones salientes del DBAdapter.

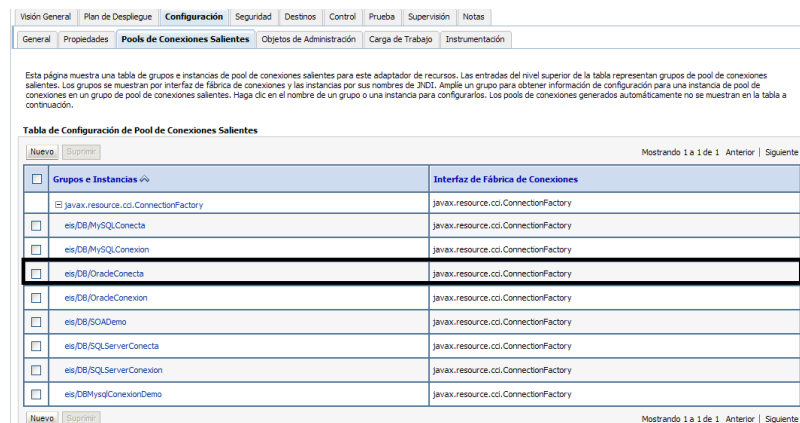


Figura 4.64 Lista de Conexiones Salientes

Luego se debe agregar el JNDI del recurso creado anteriormente a la conexión saliente. Es decir, al momento de agregar la ruta **jdbc/OracleConecta** en el valor de propiedad de la conexión saliente, se debe conectar a la base de datos Oracle por medio de esta conexión saliente **eis/DB/OracleConecta**. Se guarda y todo quedaría listo para trabajar con la base de datos Oracle.

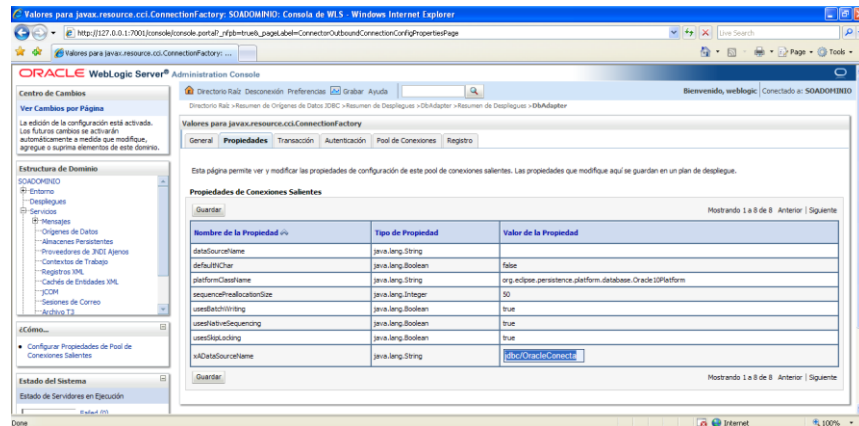


Figura 4.65 Configuración del Valor de Propiedad

Todos los pasos descritos anteriormente, fueron hechos para crear las conexiones salientes de Oracle, MySQL y SQLServer.

#### 4.5.2 Orquestación de Servicios Web

La orquestación de servicios web hace referencia a la coordinación de servicios mediante distintas invocaciones a los mismos. Para comenzar a orquestar los servicios de una Solicitud de Crédito Comercial, se debe crear un nuevo proyecto. Para este caso de demostración se lo llamará SOLICITUDCREDITOS. La creación de los proyectos para la orquestación de servicios web se debe hacer en JDeveloper, ya que es una de las herramientas para desarrollar este tipo de proyectos y que se seleccionó para construir el sistema. (Beecher, 2011)

A continuación se detalla paso a paso la creación del proyecto SOLICITUDCREDITOS en JDeveloper.

1. Primero, se debe abrir el entorno de desarrollo JDeveloper y dar click en **File** en la barra de menú y se selecciona **New**, ahí aparecerá una ventana para escoger el tipo de tecnología a utilizar.
2. Elegir la tecnología que será aplicada a este proyecto. Para este caso hay que ubicarse en la pestaña **All Technologies** y después seleccionar la categoría **General**>>>**Aplicaciones** y luego el ítem **Generic Application** y dar click en OK.

3. Después, el asistente pedirá el ingreso del nombre de la aplicación, el directorio donde se guardará y un nombre que se le da al archivo que empaquetará toda la aplicación.

4. En la siguiente ventana hay que ingresar el nombre del proyecto y luego se debe ir a la pestaña *Project Technologies* para seleccionar la tecnología SOA y agregarla en la parte derecha.

5. Luego, el asistente mostrará un lista de plantillas prediseñadas para escoger, por lo que se seleccionará la plantilla vacía y se dará click en **Finish**.

Para continuar con el proyecto se creará o importará un esquema XSD, el cual servirá para definir los tipos de elementos XML. Este tipo de esquemas se los puede crear mediante asistentes o también se puede utilizar los esquemas que se crean al momento de agregar a una plantilla el componente *DatabaseAdapter*.

En la siguiente imagen se ve que en la parte izquierda hay una lista de esquemas que se han creado de acuerdo al número de *DatabaseAdapter* que se agregan a la plantilla de componentes.

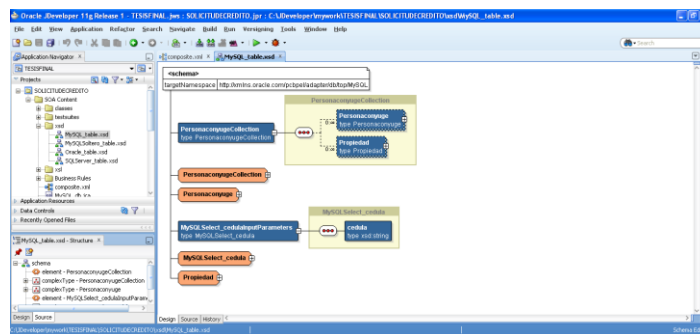


Figura 4.66 Creación del Esquema XSD

La creación de los *Database Adapter* se muestra a continuación.

### Creación de un *DatabaseAdapter*

Para crear una conexión a la base de datos local o remota, se debe arrastrar el componente *DatabaseAdapter* de la paleta de componentes a la plantilla de componentes. (May, 2012)

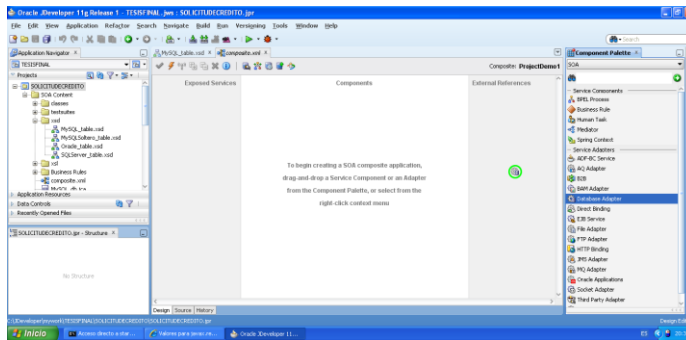


Figura 4.67 Creación de un DatabaseAdapter

Una vez que se arrastra y suelta el componente de base de datos, se desplegará una ventana tipo *wizard* que mostrará los pasos que hay seguir para que la conexión sea exitosa. Los pasos a seguir se detallan a continuación.

Al principio solo aparecerá una ventana de bienvenida, por lo que se debe dar click en siguiente para ir a la ventana que se va a ingresar el nombre del Servicio.

Una vez que se da click en siguiente, se debe crear el enlace a la base de datos. En este caso se creará un enlace a la base de datos MySQL. Para ello, se da click en la cruz para crear una nueva conexión.

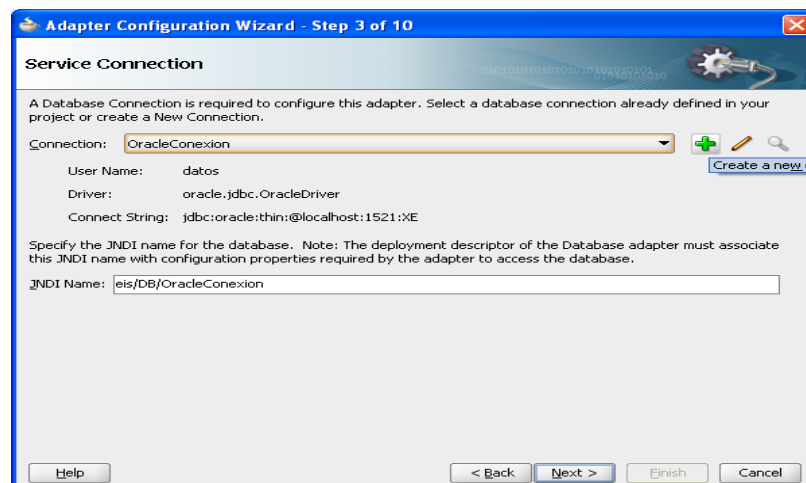


Figura 4.68 Creación de la Conexión con MySQL

Una vez que se da click en nueva conexión, aparecerá una ventana que servirá para configurar los parámetros de conexión a la base de datos como se puede ver en la siguiente figura.

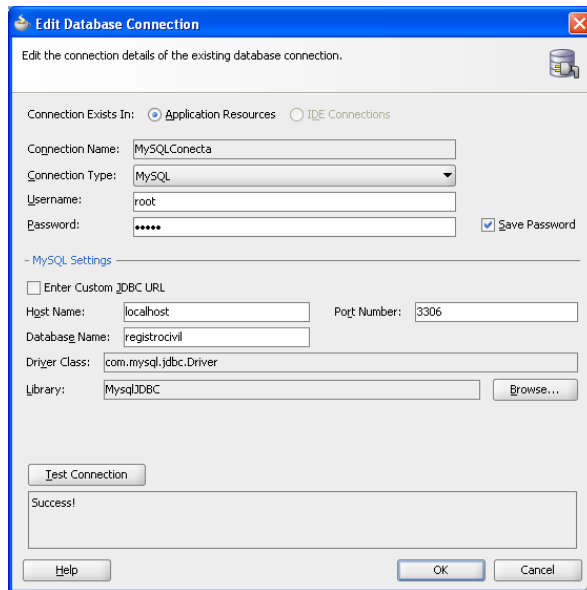


Figura 4.69 Configuración de los parámetros de Conexión de MySQL

Si todo está bien, el servicio de conexión se detallará en una ventana. Por tanto, se podrá ver que se carga el nombre de la conexión, el usuario, el driver y la dirección del servidor de la base de datos. Luego, se debe ingresar exactamente la ruta del JNDI que se creó anteriormente en la sección Estructura de Dominio. Esta ruta ayudará a enlazar el pool de conexiones JDBC. Después, se da click en siguiente y se escoge el tipo de operación que se quiere hacer en la base de datos.

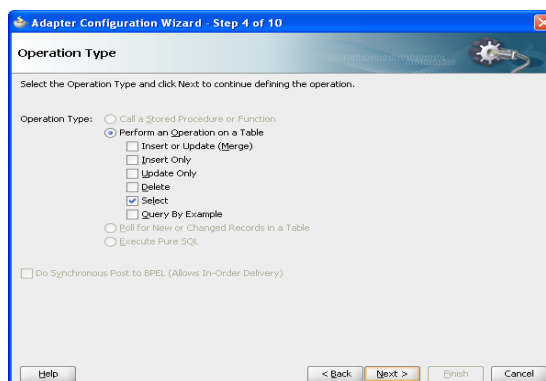


Figura 4.70 Selección de operación para la base de datos

Dar click en importar tablas para realizar las operaciones. Entonces, se visualizarán todas las tablas de la base de datos conectada. Para mostrar las tablas se da click en *Query* y luego se elige la tabla o vista para agregarla a la derecha, en donde va a ser manipulada por los tipos de operaciones. Después, click en ok.

En la siguiente ventana se selecciona los campos que se van a utilizar en las operaciones de bases de datos, por lo que se debe definir el criterio de selección, ya sea con parámetros o sin parámetros. En el primer caso que puede ser con parámetros se añadirá un nombre del parámetro a utilizar. Esto se hace dando click en el botón *add* para ingresar el nombre, el cual se añadirá en el campo *Parameters*. En segundo caso si es sin parámetros, las consultas se ejecutarán sin ningún dato para comparar en la consulta SQL.

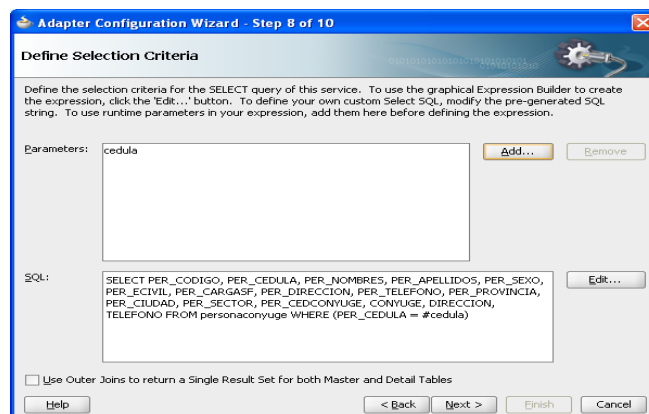


Figura 4.71 Definición del criterio de Selección

Luego se configura el SQL para introducir en la consulta el parámetro creado. Esto se hace dando click en añadir, para que se muestre la ventana de construcción de expresión. Una vez que esté lista la consulta, dar click en siguiente y luego en el botón **Finalizar**.

Estos pasos se pueden hacer para construir los servicios necesarios de base de datos.

Después de crear los servicios y el esquema, se debe arrastrar y soltar un componente llamado BPEL Process. Este componente es el orquestador de todos los servicios que estén dentro de la plantilla de componentes.



Luego de haber agregado el componente BPEL Process, se debe configurar, ya que será el orquestador de servicios de las solicitudes de crédito comercial.

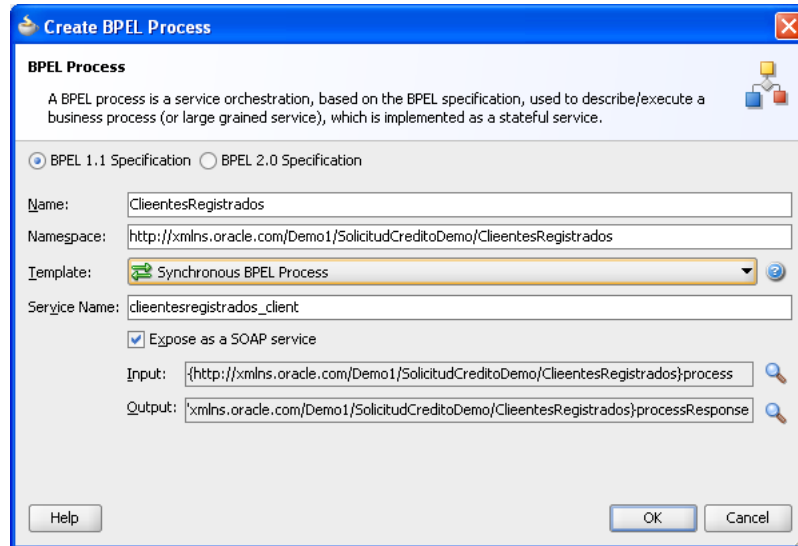


Figura 4.72 Configuración de Bpel Process Manger

Luego de haber añadido todos los componentes que se utilizarán en la orquestación de solicitudes de crédito comercial, se debe enlazar el BPEL Process, que en este caso se lo llamó **ClientesRegistrados**.

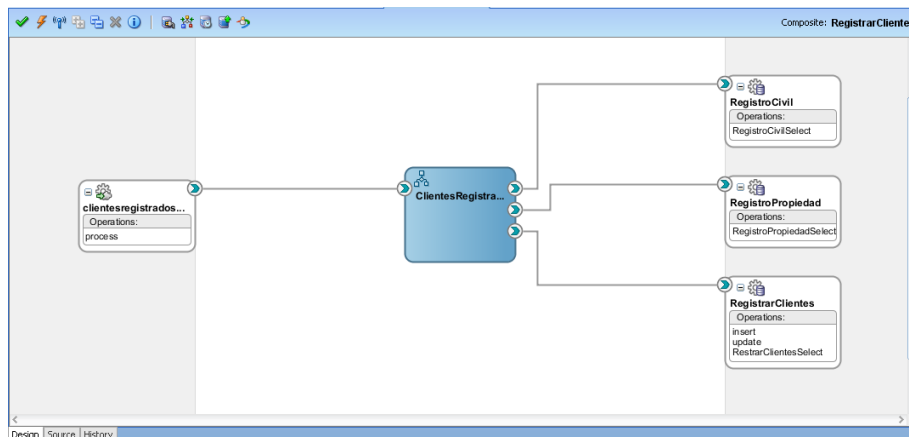


Figura 4.73 Enlace de Orquestación de Servicios

Entonces, se da doble click sobre el Orquestador de Servicios **ClientesRegistrados**, para ingresar a la parte interna.

En consecuencia, se visualizará la siguiente muestra de la estructura de una orquestación de servicios web, la cual está construida por medio de archivos WSDL, *partner link types*, *port types*, *operations*, *activities* y mensajes que son necesarios para definir las actividades de proceso.

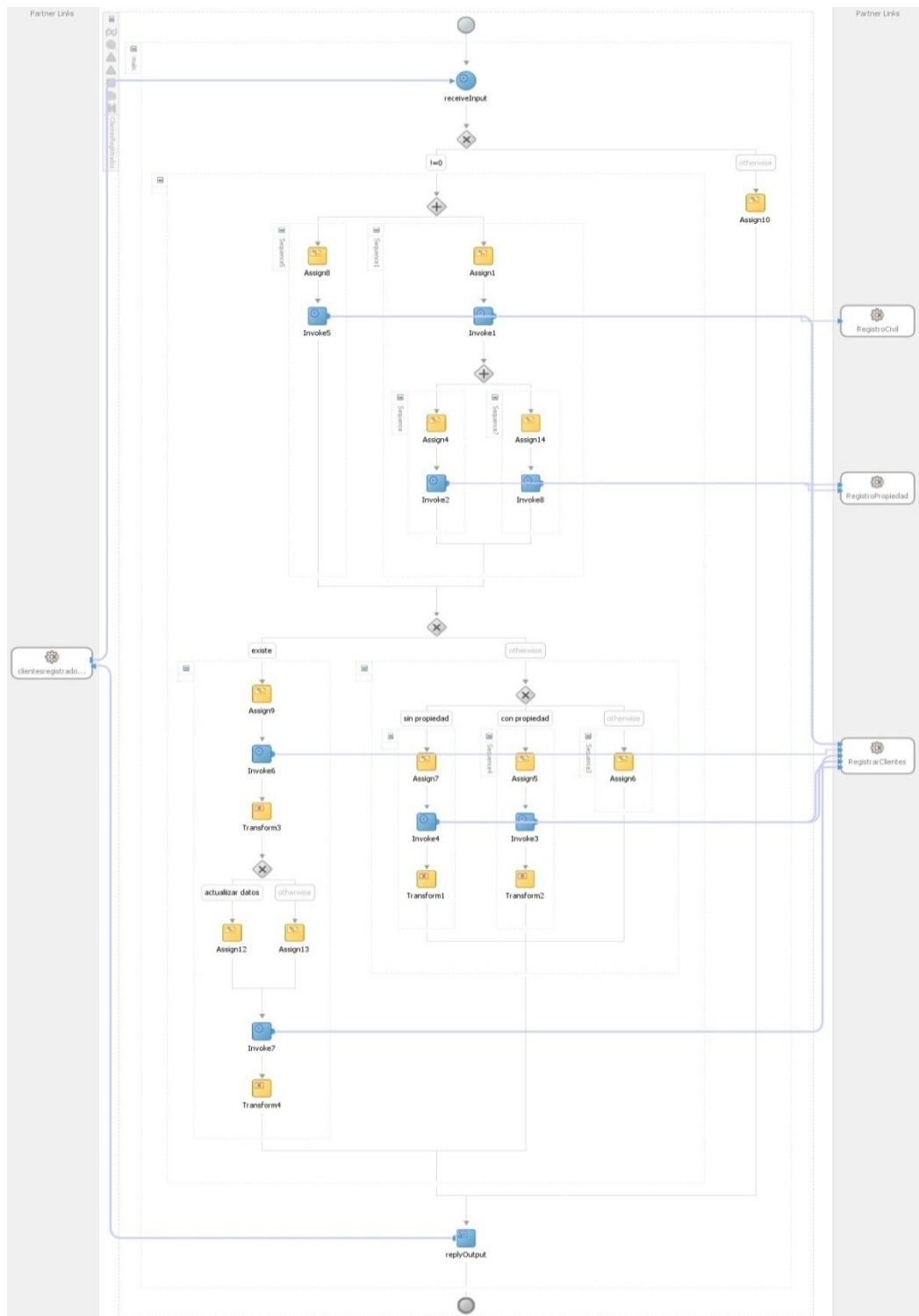


Figura 4.74 Estructura de una orquestación de Servicios web

### 4.5.3 Servicios de Transformación en BPEL

Los servicios de transformación ayudan a convertir una estructura de documento en otra, por lo que es muy importante al momento de tener un dato de entrada, que es requerido por otro servicio, pero que no son compatibles con la estructura original. En este proyecto es bastante necesario contar con este tipo de tecnologías, ya que se va a integrar estructuras de datos de tres bases diferentes. (Beecher, 2011)

Para invocar los *web service* existentes, se deben enlazar los *Partner Links* con los servicios web dentro del proceso principal. Es decir, se debe agregar 4 actividades *invoke* en el flujo del proceso principal, las cuales se enlazarán con los *partnerlinks* creados. Cada invocación a un servicio es identificado por un *PartnerLink*, y para que suceda esto, el *partnerlink* solicita una dato de entrada y otro de salida.

Una vez que se enlazan las actividades con los *partnerlinks* se debe empezar a manipular los datos que están dentro del flujo de procesos. Es decir, hay que empezar a mapear las variables de entrada y salida de los servicios web expuestos.

Para realizar la manipulación de los datos, se agregará al flujo de procesos, la actividad *Assign*. Esta actividad permite copiar contenido de una variable a otra (*from-to*). En la siguiente imagen se ve como se asigna la variable de entrada **cédula** con el parámetro de entrada de la consulta a la base de datos MySQL.

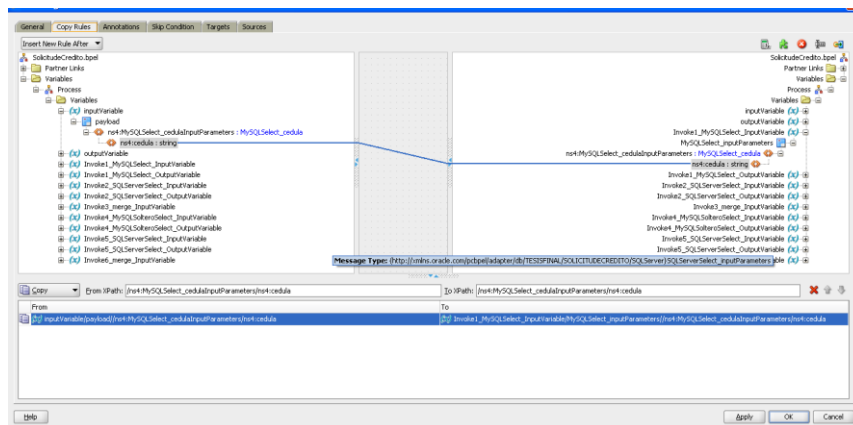


Figura 4.75 Manipulación de datos con la actividad Assign

En la segunda figura se muestra como se copia el contenido de las variables de salida de la consulta MySQL a las variables de salida del BPEL Process.

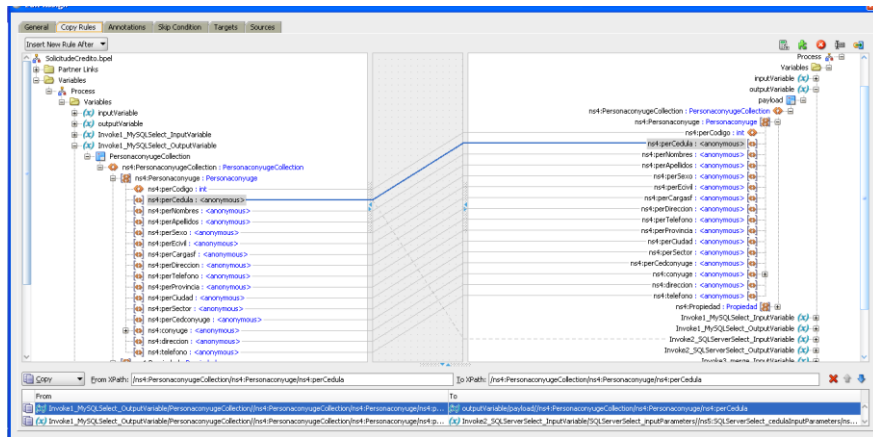


Figura 4.76 Copia de datos a una variable de salida

En la tercera figura se visualiza como se asigna la variable de salida cédula de la consulta MySQL con el parámetro de entrada de la consulta a la base de datos SQLServer .

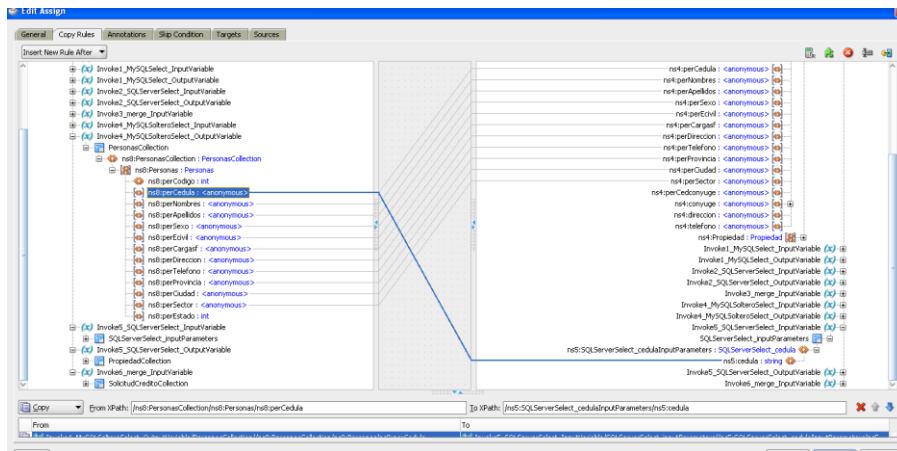


Figura4.77 Asignación de datos entre variables

En la cuarta imagen se verá cómo se copia el contenido de las variables de salida de la consulta SQLServer a las variables de salida del BPEL Process.

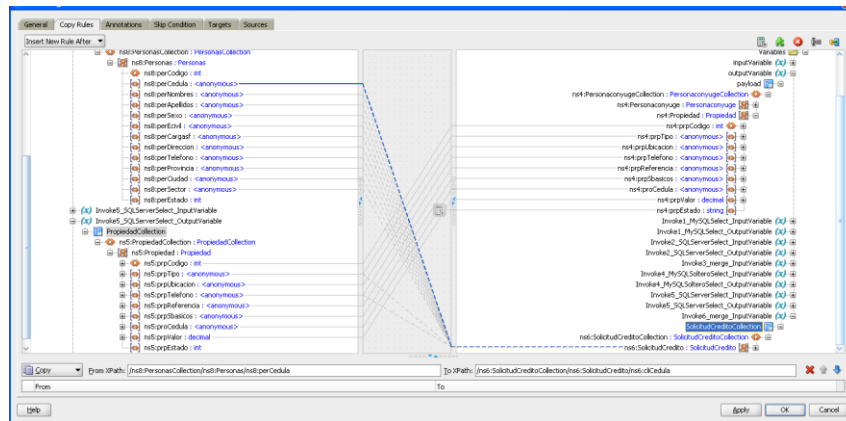


Figura4.78 Copia de datos entre variables de dos Base de Datos

Finalmente, se copian todas las variables de salida de BPEL Process a las variables de entrada del *web services* expuesto de la base de datos Oracle.

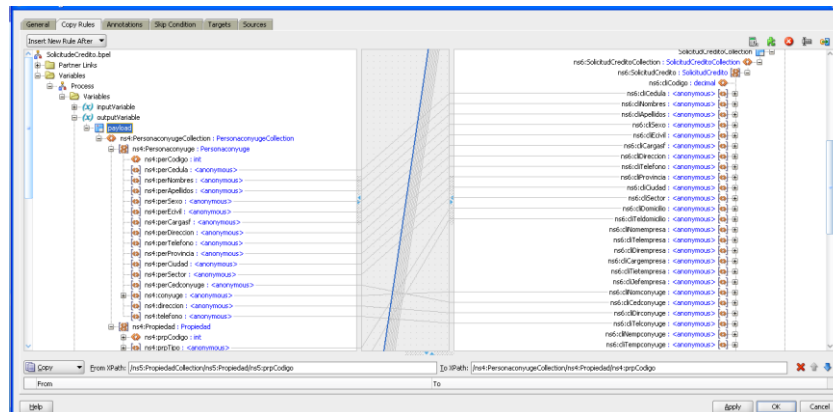


Figura4.79 Asignación de datos entre variables de salida de dos Base de Datos

#### 4.6 Invocación y Accesos a un Servicio BPEL a través de Adaptadores

Para que los Sistemas Empresariales puedan interactuar con los procesos de negocio, es necesario emplear los servicios de adaptadores, los cuales constituyen una forma de integración entre aplicaciones y servicios que mejoran la eficiencia operacional. También, permiten que BPEL se adapte dinámicamente a cambios en las aplicaciones empresariales, sin necesidad de recurrir a cambios en la codificación interna, ya que los servicios adaptadores se basan en estándares que implementan conectividad bidireccional entre las aplicaciones de Oracle y los Sistemas Empresariales. (Oracle, 2010)

Estos adaptadores soportan los siguientes estándares abiertos.

- WSIF: Web Service Invocation Framework.
- J2CA: J2EE Connector Architecture.
- WSIL: Web Service Inpection Language.
- WSDL: Web Service Description Language.
- JMS: Java Message Service.

Los estándares J2CA y WSDL son utilizados por el *framework* de adaptadores para integrar transparentemente el adaptador de recursos J2CA con Oracle BPEL Process Manager. Los adaptadores se categorizan en:

**Adaptadores de Entrada:** Sirven para que las comunicaciones asíncronas estén dentro de los servicios de notificación de eventos. Estos adaptadores escuchan o consultan los cambio en los eventos (*back - end*). Cuando está en ejecución, este adaptador se registra como un escuchador para la aplicación (*back - end*) que está ajustada para enviar eventos al adaptador, por lo que también puede consultar a la aplicación (*back - end*), que por lo general es una base de datos o un archivo por cada evento que requiere la aplicación. **(May, 2012)**

**Adaptadores de Salida:** Para procesar servicios síncronos solicitud-respuesta. Estos adaptadores reciben solicitudes de los clientes, las transforman al formato nativo de datos (*back - end*) y llaman al método adecuado en la aplicación (*back - end*). Luego, recupera la respuesta (*back - end*) y la pasa al *framework* de adaptadores después de haber realizado un traducción inversa. **(May, 2012)**

**Adaptadores de Base de Datos:** Soportan una gran variedad de motores de base de datos como por ejemplo: Oracle, MySQL, SQLServer, BD2, Infomix, SyBase, etc. También, soporta cualquier base de datos que sea compatible con los estándares JDBC 1.x y JDBC2.x. Estos adaptadores pueden ser usados para hacer consulta de datos (*Select*), realizar manipulación de datos (*Merge, Insert, Update y Delete*) y para invocar procedimientos almacenados. **(May, 2012)**

Para invocar un adaptador de base de datos como un servicio, primero se debe especificar la variable de entrada en la actividad *invoke*, la cual especifica la acción (*Select, Merge, Insert, Update, o Delete*) en el *PartnerLink*, que a la vez se ejecutará al momento de realizar una solicitud. Las operaciones seleccionadas en el *PartnerLink* son invocadas y almacenadas en un archivo WSDL que se genera para interactuar con el adaptador de base de datos. Este tipo de servicios puede ser invocado por cualquier cliente que maneje la semántica de invocación del *web services*. (May, 2012)

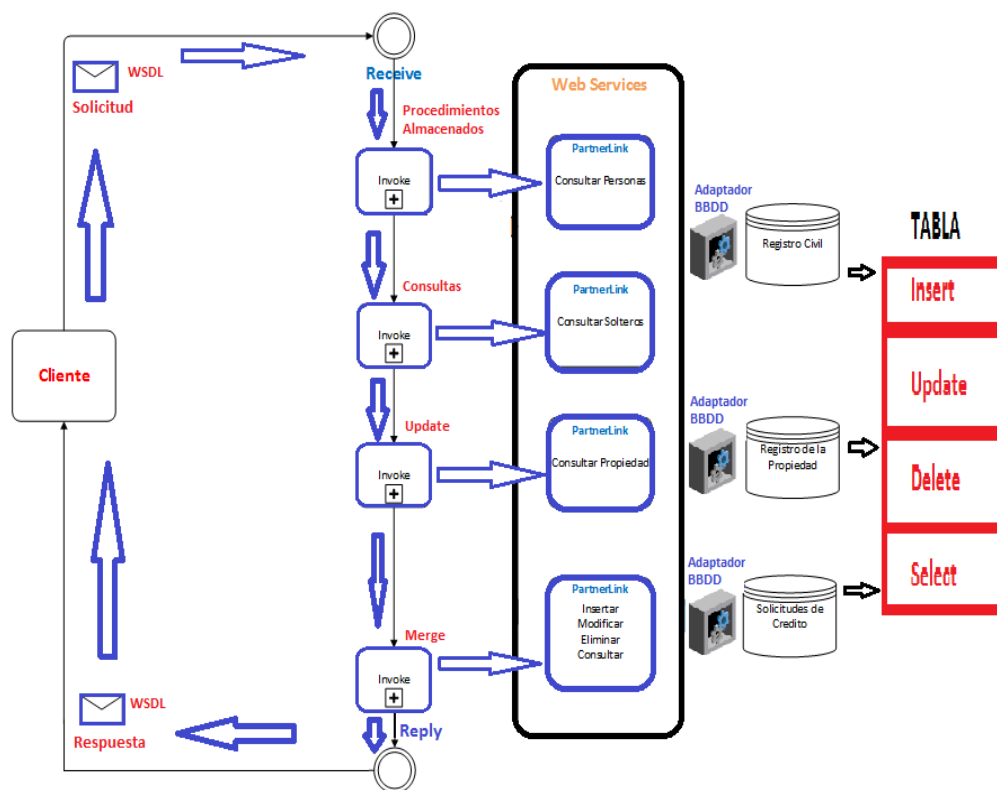


Figura 4.80 Invocación a un adaptador de Base de Datos

#### 4.7 Procesamiento de los Servicios de Notificación

El proceso de servicios de notificaciones permite alertar y enviar un mensaje a los usuarios que tengan pendientes aprobaciones de distintas solicitudes en el sistema. Los mensajes que se procesan pueden ser enviados mediante correo electrónico, mensaje de voz o un SMS.

Este proceso de servicio de notificaciones soportan el envío de correos electrónicos utilizando el protocolo SMTP y la recepción de mensajes desde cuentas de correo electrónico que estén basadas en IMAP y POP. De esta forma el usuario tiene la facilidad de responder de inmediato una solicitud. **(Oracle, 2009)** Por ejemplo, en el caso de una solicitud de crédito comercial, el flujo de trabajo envía un mensaje vía correo electrónico al gerente comercial, para aprobar o rechazar la solicitud de crédito comercial del cliente. El gerente deberá aprobar o rechazar la solicitud del cliente respondiendo el mensaje con la opción apropiada.

En conclusión, se puede decir que las herramientas utilizadas para la orquestación de servicios web son bastante dinámicas, ya que permite interactuar con varias tecnologías a la vez. Es por eso que se debe tener claro que la parte de estructuración de base de datos no se puede dejar de lado, porque al momento de interactuar con distintas bases de datos, estas deben estar bien formalizadas para que los esquemas de datos se creen sin ningún problema y así, la manipulación de datos en las variables de entrada y salida sean confiables.



## CAPITULO 5

### VALIDACIONES Y PRUEBAS

En este capítulo se resaltaré el análisis de una solicitud de crédito comercial, por lo que se determinará como se debe ingresar y consultar la información de un crédito comercial. Se describirá la forma automática y manual de validar la información, para que una solicitud de crédito sea aprobada, rechazada o que tenga un estado pendiente. Por último, se estudiará cómo interactúa la consola de administración, en el momento que se despliega un proceso de orquestación en cada instancia.

#### 5.1 Análisis y Registro de una Solicitud de Crédito Comercial

##### 5.1.1 Análisis de una Solicitud de Crédito

El análisis de solicitudes describe los procesos que van a intervenir para interactuar con la información del solicitante. Esta información puede estar almacenada localmente o remotamente.

##### Consulta de Información del Cliente

En este proceso se consulta la información del cliente que se encuentra almacenada en la base de datos de la entidad que brinda estos servicios. Para realizar esta consulta (*call service*) el orquestador **CientesRegistros** utilizara su variable de **entrada** para recibir la solicitud, en este caso sería la **cédula del cliente**. Por lo que, si la cédula existe se obtendrá la información del cliente en una variable de **salida**.

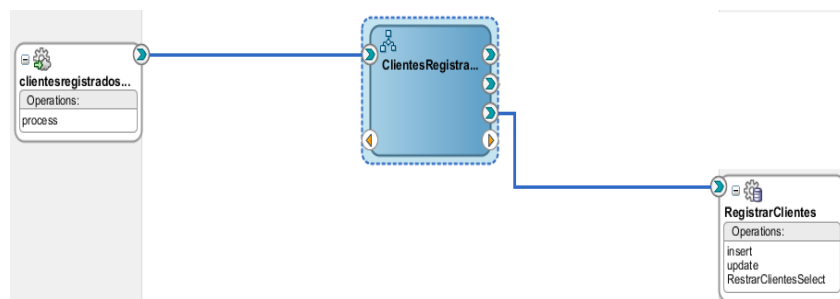


Figura 5.1 Proceso de consulta de Información del Cliente

El proceso de llamado que el orquestador recibe es a través de un puerto llamado **registrarclientes**, el cual está listo para escuchar a cada solicitud.

Una vez recibida la solicitud, la actividad *assign* se encarga de mapear la información a sus *partners* respectivos, lo cual consta de **copiar** la información de **entrada** a **entrada** para obtener la **salida** o **respuesta**, que en este caso sería la información del cliente.

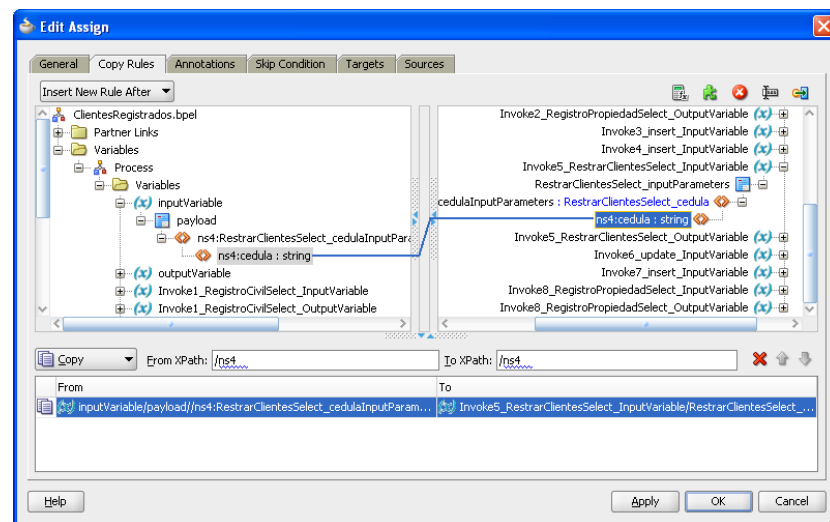


Figura 5.2 Mapeo de Variables

### Solicitud en curso

En este proceso se valida si la información que se envía es correcta, en este caso se validará si la cédula que se recibe es correcta. Para validar la información que entra, se utiliza la actividad *Switch*, que sirve para realizar todo tipo de operaciones condicionales. Esta actividad contendrá todas las actividades *assign* involucradas en el proceso de consulta de información del cliente. Esto hará que la información de entrada y salida sea válida en cada proceso de solicitud en curso.

También, esta actividad validará el estado civil y el tipo de propiedad de cada cliente, para tener una información más específica de cada cliente.

Por tanto, este proceso devolverá la información del cliente solo si la variable de entrada recibe una cédula correcta, de lo contrario devolverá un mensaje de advertencia como: “Ingrese una cédula válida”.

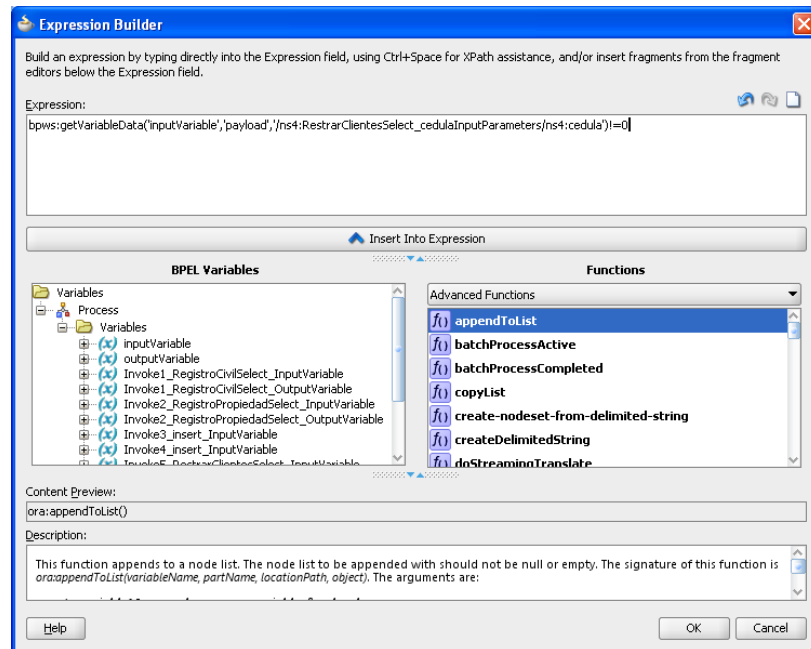


Figura 5.3 Manipulación de Variables de entrada

## Importación de la Información del Cliente

Se llama a este proceso en el caso de que la información del cliente no esté en la base de datos local de la entidad de solicitudes de crédito o cuando la información en la misma este desactualizada. La información que se solicitará será de la base de información del Registro Civil.

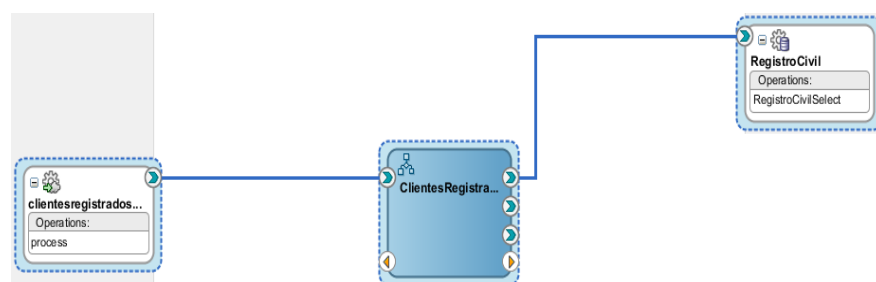


Figura 5.4 Invocación a la base del Registro Civil

En este caso el proceso de llamado que el orquestador recibe es a través de un puerto llamado **RegistroCivilSelect**, el cual invocará a las consulta de información del cliente en la base de datos del registro civil. Para efectos de prueba, la información del Registro civil será importada desde un sistema externo, el cual esta creado en php y la información que se almacena es una base de datos MySQL.

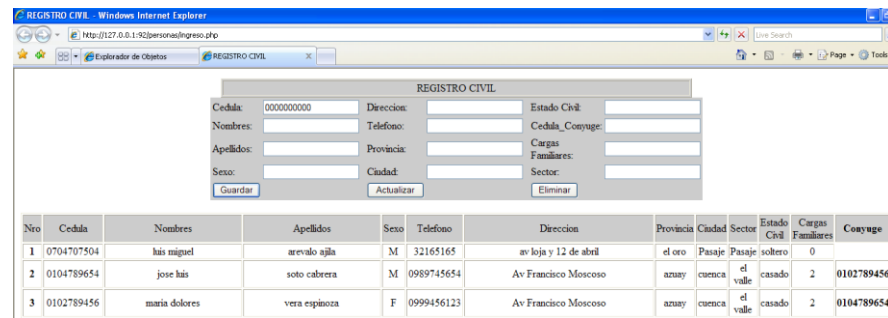


Figura 5.5 Verificación de la Información del Registro Civil

De igual manera hará esto para otra base de dato externa del registro de la propiedad. En esta base se extraerá la información de la propiedad que pueda tener el solicitante.



Figura 5.6 Invocación a la base del Registro de Propiedad

En este caso el proceso de llamado que el orquestador recibe es a través de un puerto llamado **RegistroPropiedadSelect**, el cual invocará a las consulta de información del cliente en la base de datos del registro de la propiedad. Para efectos de prueba, la información del Registro la Propiedad será importada desde un sistema externo, el cual esta creado en Visual Basic .Net y la información se almacenará en una base de datos SQL Server 2008.

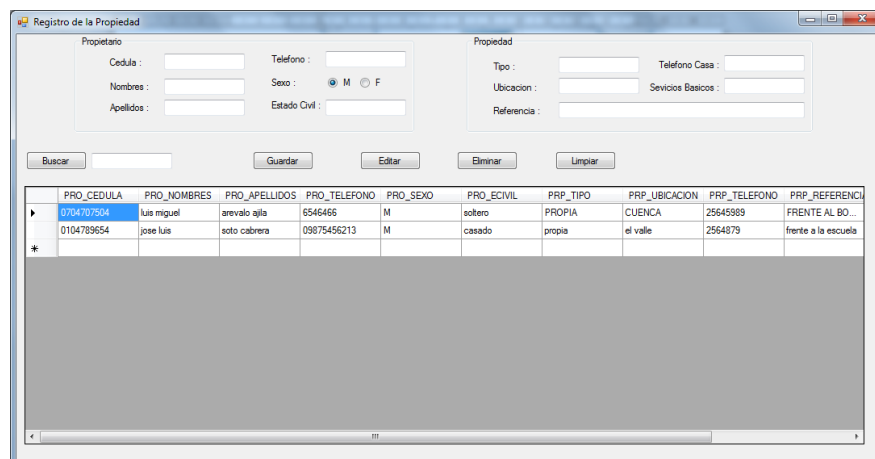


Figura 5.7 Verificación de la Información del Registro de Propiedad

### Capacidad de endeudamiento

En este caso el sistema comprueba por medio de un cálculo el porcentaje de endeudamiento del solicitante. **(Tucker A, 1976) (Van Horne, 1995).**

$$\text{RAZON DE ENDEUDAMINETO} = \frac{\text{PASIVO TOTAL}}{\text{ACTIVO TOTAL}} = \%$$

Este cálculo se ejecuta internamente al momento de validar los datos referentes a los ingresos y egresos que puede tener el cliente.

### Capacidad de Pago

En este caso se comprueba el valor de las cuotas de pago que el cliente debe hacer para cancelar el total del crédito comercial solicitado, el cual dependerá de la entrada que el cliente vaya a abonar y del número de meses que se dé el crédito, como se muestra a continuación. **(Tucker A, 1976) (Van Horne, 1995).**

$$\frac{\text{Total de Crédito} - \text{Valor de Entrada}}{\text{Número de Cuotas}} = \text{Valor de Cuotas}$$

Este valor servirá para luego poder comparar con la tasa de endeudamiento que el cliente tenga y así poder asignar el tipo de aprobaciones que puede tener un cliente, de acuerdo al tipo de política que tenga la empresa.

### **Estados de Solicitudes de los Clientes**

Los tipos de **Estados** dependerán de los distintos factores que el cliente tenga como garantía y le sirvan para obtener un crédito comercial. Para efectos de prueba se implementó en este proyecto una propia política de **Aprobación del Crédito, Rechazo del Crédito y Créditos Pendientes**.

Para lo referente a las Aprobaciones de Crédito, el cliente deberá aprobar la **capacidad de endeudamiento** y tener una **casa propia**.

Con respecto a lo que son Créditos Pendientes, este tipo de estados se asignaran a los clientes que no aprueben el nivel de **capacidad de endeudamiento** o que no tengan **casa propia**. Si este estado está asignado a una solicitud de un cliente, el Gerente de ventas deberá tomar una decisión de **Aprobado** o **Rechazado**, dentro de un plazo no mayor a los 7 días.

Para lo referente a **Rechazo de crédito**, solo se rechazara a los clientes que estén con el estado **Pendiente** y que no se haya dado una respuesta dentro de los 7 días. Por ejemplo, si es que tenía casa propia, pero al momento de pedir un crédito comercial el sistema muestra que no tiene registro de propiedad. El cliente tiene la oportunidad de ir a registrar su propiedad para que el sistema pueda actualizar los datos del cliente y pueda aprobarse el crédito.

## 5.1.2 Verificación de la Información

### Validaciones de información del Cliente

En esta etapa se validará todos los campos en donde el usuario pueda agregar información sin ningún tipo de errores de escritura. Además, se validará los campos que no sean nulos, que tengan solo número o solo letra y por último que al momento de guarda la información todos los campos hayan sido llenados.

También, se verificará que el nivel de endeudamiento del cliente este dentro de los márgenes de endeudamiento que la empresa tiene por política. En este caso para efectos de prueba se validará que la tasa de endeudamiento sea mayor al valor de la cuota que pagara dentro del número de cuotas establecidas. Esto se ejecutará en el momento de ingresar la información de la solicitud de crédito.

### Toma de decisiones

Este proceso se encargará de actualizar la información de las solicitudes de crédito que hayan sido almacenadas como pendientes y que sobrepasen los 7 días. Para ejecutar este tipo de operaciones se implementará un proceso de trabajo en la base de datos que almacene la información de las solicitudes de crédito comercial. Este proceso de trabajo estará constantemente monitoreando los cambios que puedan tener los factores de información que están involucrados en la aprobación o rechazo de una solicitud de crédito.

### Estado de decisiones

Este proceso será accionado propiamente por el gerente de ventas, ya que tendrá la posibilidad de cambiar el estado de la solicitud de crédito a **Pendiente**, **Rechazado** o **Aprobado**. En el caso de querer ejecutar esta tipo de acción, el gerente deberá consultar o buscar primero la solicitud que quiere modificar y luego dar click en el **Nro. De Solicitud**, para luego proceder a la **Toma de Decisión**.

Una vez que escoja una decisión, el gerente en este caso deberá aceptar su acción, para que así pueda tener efecto la toma de decisión, de lo contrario, debe cancelar para que su decisión no tenga efecto en la solicitud consultada. En caso de aceptar cualquier decisión, se guarda como una solicitud grabada **Manualmente**, ya que todas las solicitudes que se guardan estarán con el tipo de Decisión **Automática**.

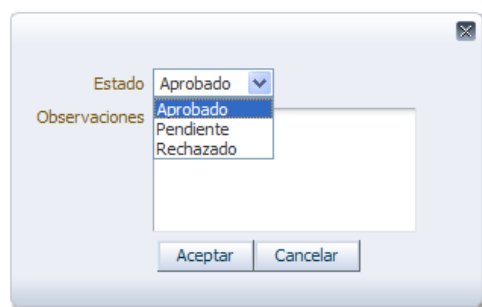


Figura 5.8 Estados de Decisiones

### 5.1.3 Creación de la Información

#### Registro de Solicitudes de crédito

Una vez que esta validado todo el sistema guardara la información de la solicitud de crédito comercial solicitada por el cliente. Hecho esto, se almacenara en la base de datos de la empresa y estará dispuesta para realizar cualquier modificación de la misma, de manera interna o de manera externa si es el caso de que quieran extraer información actualizada de entidades externas.

#### Actualización de Estados

En el caso de actualizar información desde una base de datos externa, el sistema invocará nuevamente al servicio de Orquestaciones para tomar la decisión de actualizar o no los datos del cliente.



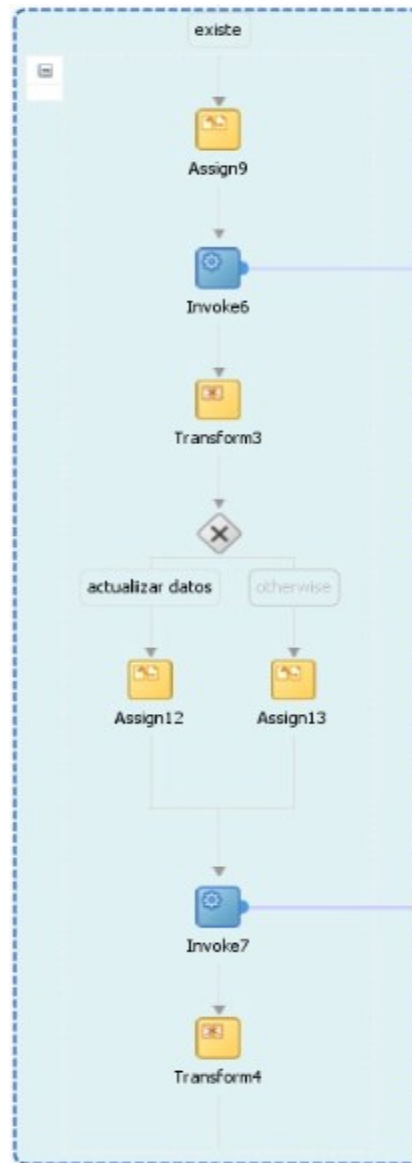


Figura 5.9 Orquestación de para la Actualización de Datos

En este proceso lo que hará el orquestador de servicios es preguntar si existe primero en la base de datos local para realizar la acción de Update de los datos. Por lo que, el usuario antiguo pasa a eliminarse con un estado lógico, lo que permitirá que su información sea consultada a futuro como un historial de cambios.

## 5.2 Evaluación de Despliegue

Después de haber desarrollado todas las aplicaciones web, componentes, bases de datos, servicios y procesos que se involucran en la creación del sistema de solicitudes de crédito, el siguiente paso será ejecutarlos para verificar su correcto funcionamiento.

Primero, se comprobará que el despliegue del proceso principal “OrquestaSolicitud”, que es el encargado de la orquestación de las solicitudes de crédito comercial, interactúe con los servicios web de las consultas a las bases de datos MySQL, SQLServer y Oracle.

Para ejecutar el despliegue se debe dar click derecho en el proyecto luego seleccionar Deploy->OrquestacionSCC. (Beecher, 2011)

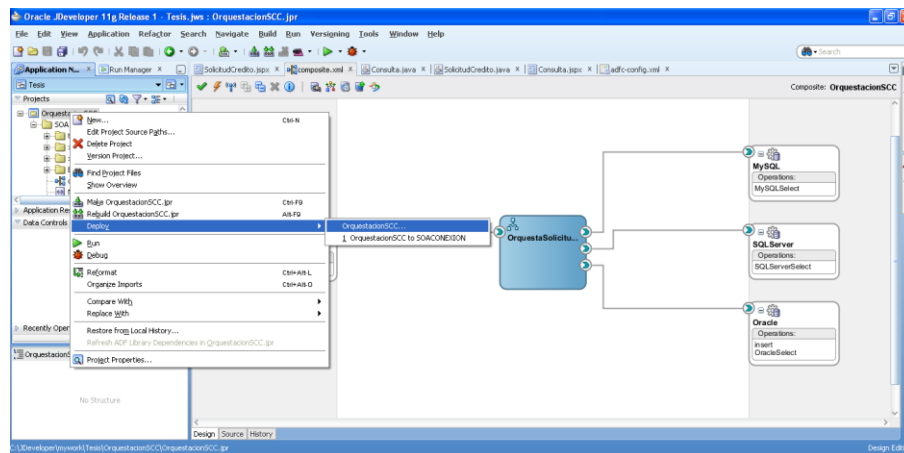


Figura 5.10 Despliegue de Procesos

Con la ayuda de un asistente se procede a crear todo el proceso de despliegue. La siguiente figura muestra en donde se debe realizar la acción de despliegue, por lo que se debe escoger el servidor de aplicaciones y luego dar click en siguiente.

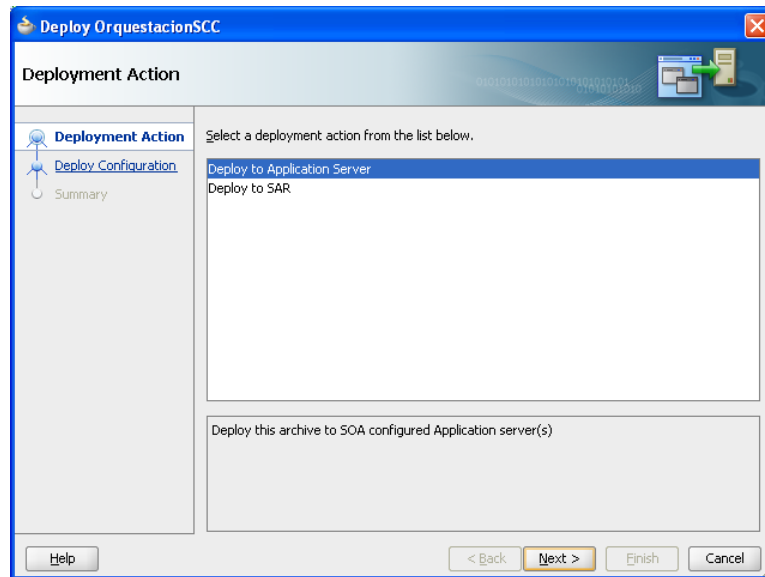


Figura 5.11 Acción de Despliegue

En la siguiente imagen se configurará el despliegue, por lo que se puede usar la opción de versionamiento del mismo, en el caso de que se realice nuevos cambios.

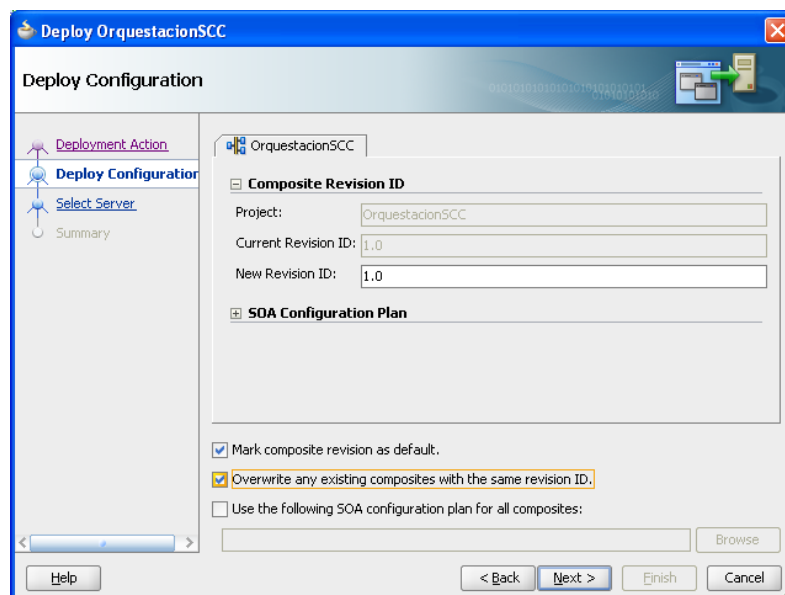


Figura 5.12 Configuración de Despliegue

En la siguiente figura se debe seleccionar el nombre del servidor de aplicaciones SOA; para el caso de demostración se creó SOACONEXION, pero si se desea crear otro se lo haría dando click en la cruz que se encuentra en la parte superior de la ventana.

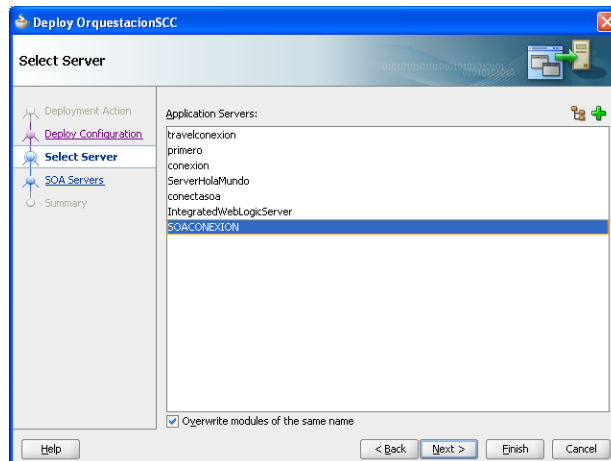


Figura 5.13 Selección del Servidor

Luego de seleccionar el servidor de aplicaciones, se elegirá un domino que se haya creado en el servidor SOA, la partición creada y también verificar si está en ejecución.

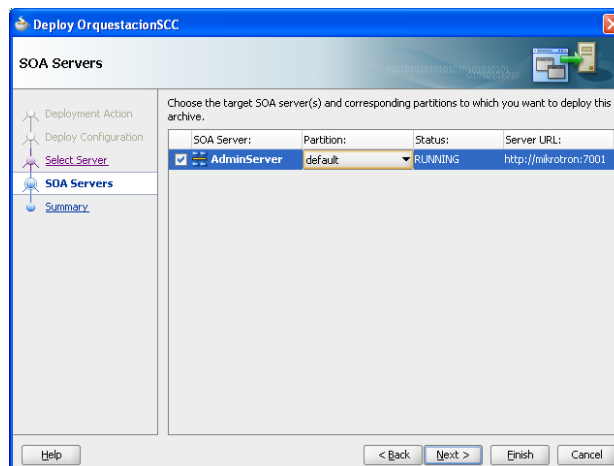


Figura 5.14 Selección del Dominio SOA

Una vez hecho todo, se verifica que todo esté bien y se da click en el botón ***Finish.***

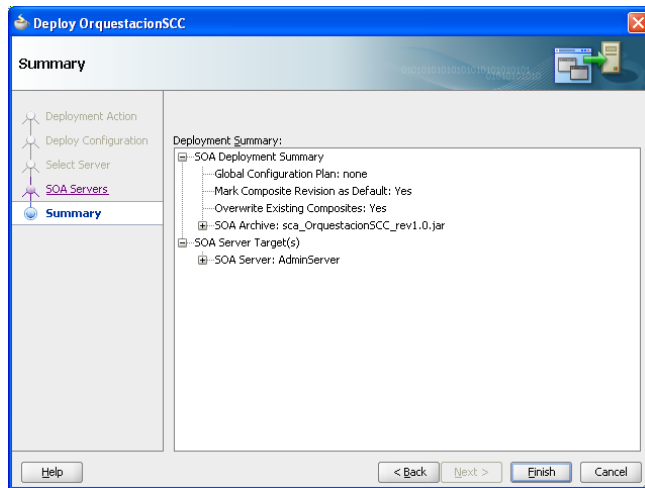


Figura 5.15 Resumen de Despliegue

El resultado que se obtiene si la aplicación se despliega correctamente es BUILD SUCESSFUL.

### 5.3 Evaluación de la Consola de Oracle BPEL

Segundo, se debe realizar las pruebas necesarias en la consola de Administración de Oracle BPEL. En esta consola, que está diseñada en un entorno web, se muestran todos los procesos que hayan sido desplegados y que están listos para ser probados. Para evaluarlos se pulsa el botón probar. (Beecher, 2011)

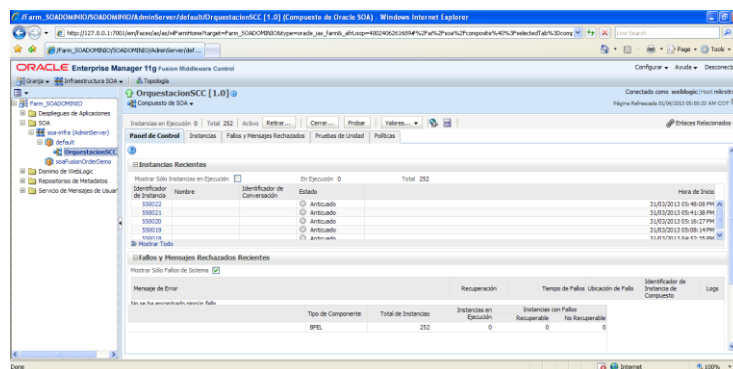


Figura 5.16 Consola de evolución de procesos

Después de dar click en probar, se accede a otra página para ingresar los datos necesarios y realizar un servicio de solicitud. Para el caso de demostración se ingresarán los datos necesarios, que permitirán verificar la respuesta obtenida, por medio de los web services que interactúan con la información almacenada en bases de datos remotas.

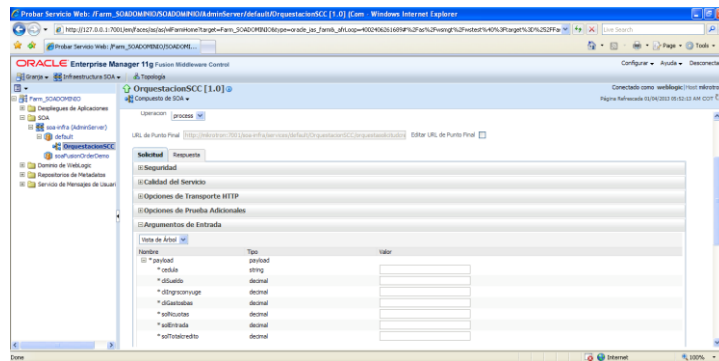


Figura 5.17 Pruebas de Evaluación

Cada respuesta de la solicitud enviada se mostrará en la segunda pestaña. Además, se utilizará el servicio web desde otras aplicaciones, ya que existe un campo WSDL que es que registra la dirección del cada servicio web que se ejecute.

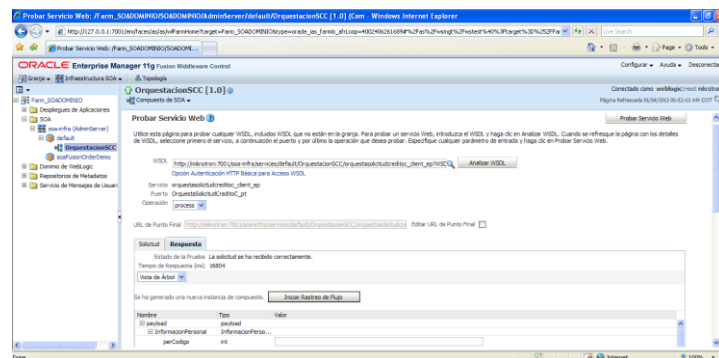


Figura 5.18 Respuesta de Evaluación

## 5.4 Auditoria de Ejecución

Siempre cuando se inicie una nueva instancia, la consola de administración permite monitorear los resultados de ejecución de las instancias, realizando así las depuraciones de cada proceso que se ejecute. (Beecher, 2011)

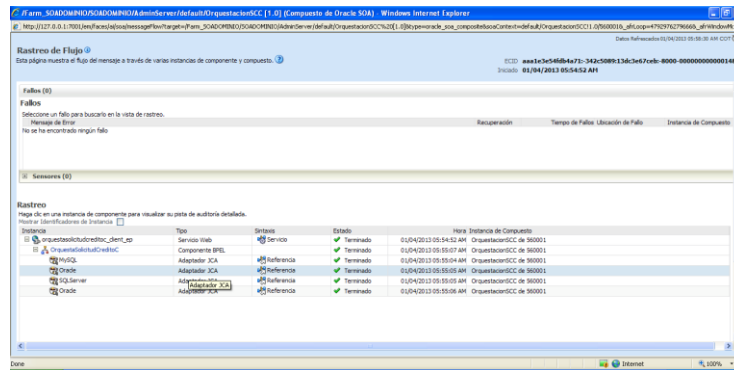


Figura 5.19 Auditoria de Ejecución de Procesos

En la siguiente imagen se puede ver los resultados de las interacciones del flujo de procesos con los servicios expuestos.

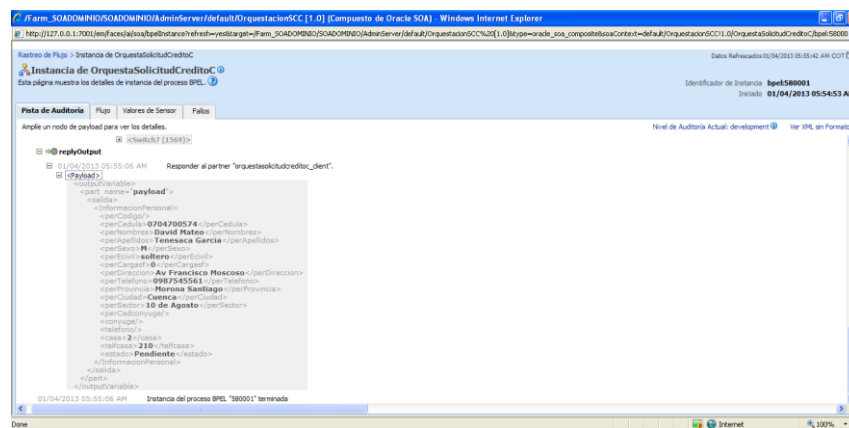


Figura 5.20 Resultados de Interacción de los procesos

## 5.5 Verificación del Funcionamiento de la Aplicación

Una vez que se pudo constatar que el proceso de orquestación de solicitudes de crédito está funcionando en la consola de administración, se debe publicar en una página web. En esta página web puede comprobar el funcionamiento de los *web services* que se crean a partir del despliegue de un proceso BPEL.

En la siguiente figura se muestra cómo se puede interactuar fácilmente con las páginas creadas mediante un *taks flow*. (Ralph Gordon, 2010).

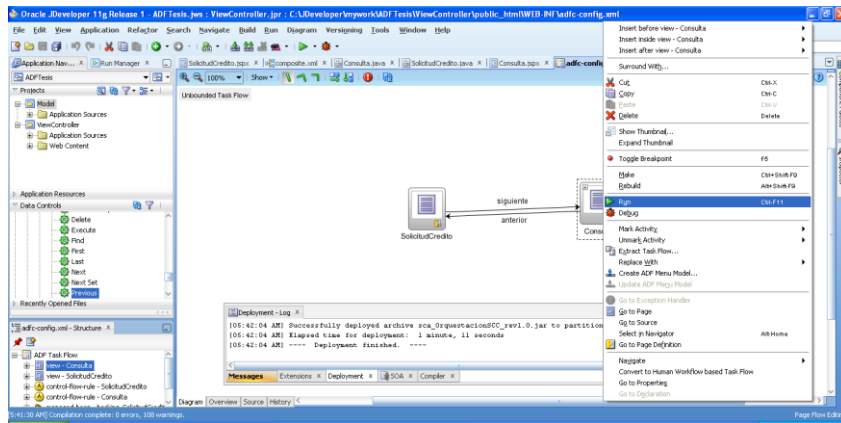


Figura 5.21 Página de Flujo de tareas

Una vez que se ejecuta la página web creada en un *taskflow*, se puede ver que está lista para interactuar con la información que se registre.

## 5.6 Verificación de Interacción con la Base de Datos

Una vez que se haya registrado la solicitud desde la página web, se debe verificar que la información se esté almacenando correctamente en la base de datos Oracle.

No. Solicitud	Fecha	Vendedor	No. Cuentas	Valor Cuentas	Entrada	Total Crédito	Cuentas	Clientes
#. Solicitud	#. Fecha	#. Vendedor	#. Solicitudes	#. Solicitudes	#. Solicitudes	#. Solicitudes	#. Solicitudes	#. Clientes
#. Solicitudes	#. Fecha	#. Vendedor	#. Solicitudes	#. Solicitudes	#. Solicitudes	#. Solicitudes	#. Solicitudes	#. Clientes

Figura 5.22 Verificación de la información en el sistemas de Orquestación

En la siguiente figura se comprueba mediante una sentencia SQL la información registrada de un cliente que solicito un crédito comercial.



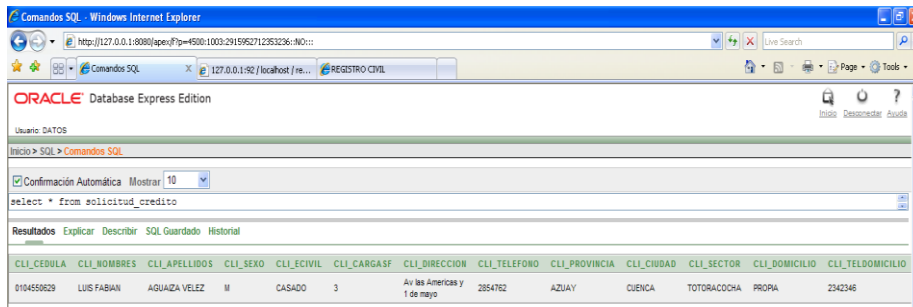


Figura 5.23 Verificación de la Información en la base de datos Oracle

## 5.7 Comprobación del Funcionamiento de Adaptadores de Bases Externas

En esta sección se comprobara que los adaptadores de bases de datos estén realizando sus procesos de ejecución de sentencias correctamente.

En la siguiente figura se detalla la información de la propiedad de un cliente, que esta almacenada en una base de datos del registro de la propiedad. Esta información esta almacenada y administrada en el gestor de base de datos SQLServer 2008.

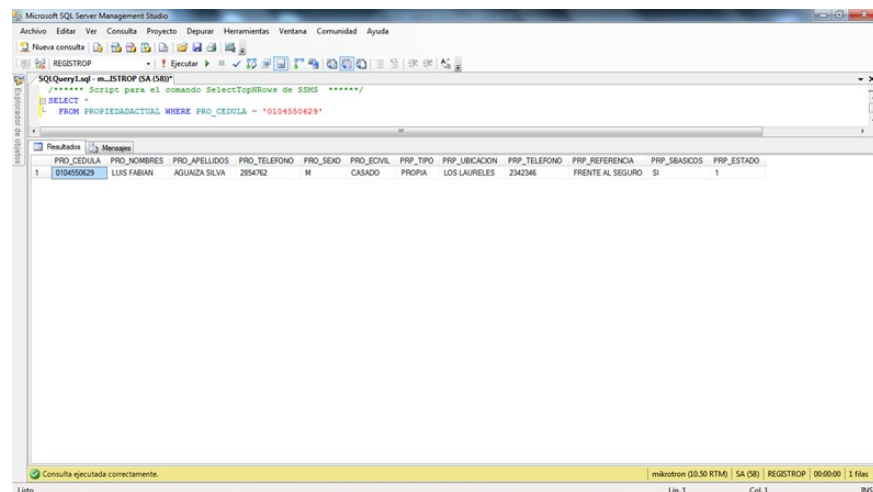


Figura 5.24 Verificación de la Información en la base de datos SQL Server

La siguiente imagen en cambio visualiza la información del mismo cliente pero en la base de datos del registro civil. Esta información esta almacena y gestiona por MySQL.

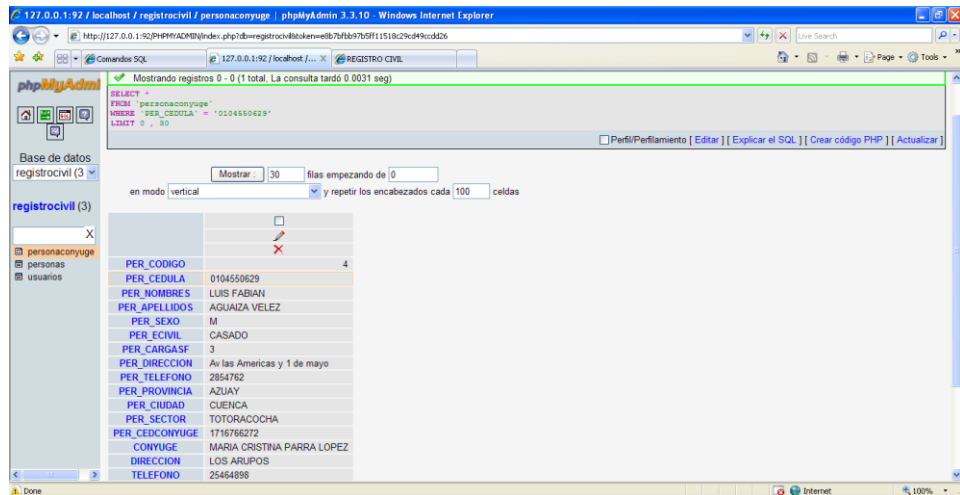


Figura 5.25 Verificación de la Información en la base de datos MySQL

## 5.8 Prueba del Servicio de Notificación

Para comprobar los servicios de notificación, se ha realizado una consulta directamente a la base de datos. Esta notificación es almacenada dependiendo de la información que se obtenga del cliente. La información obtenida es orquestada por *BPEL process* y luego es condicionada mediante las actividades que se involucran para cada tipo de casos. Los casos que pueden darse son:

**Aprobado:** Este caso se da cuando los ingresos del cliente superan el monto del crédito solicitado y el cliente tiene casa propia. En este caso el tipo de decisión tomada sería automáticamente por el sistema, ya que las reglas de negocio se crearon internamente en el Orquestador de Procesos.

**Pendiente:** Este caso se da cuando los ingresos del cliente son menores al crédito que está solicitando y el cliente tiene casa propia o puede ser viceversa. En este tipo de caso el gerente comercial puede tomar la decisión de aprobar o no el crédito.

**Rechazado:** Este caso se da cuando los ingresos del cliente son menores al monto del crédito solicitado y el cliente no tiene casa propia.

En este caso el tipo de decisión tomada sería automáticamente por el sistema, ya que las reglas de negocio se crearon internamente en el Orquestador de Proceso “OrquestaSolicitud”.

En la siguiente figura se muestra el registro de un cliente cuando el estado del crédito es aprobado.

The screenshot shows a web application window titled "Nueva Solicitud de Crédito". At the top, there are buttons for "Grabar", "Nueva Solicitud", "Buscar", and "Ir a Solicitudes". Below the title bar, there are tabs for "Datos Personales", "Información Financiera", "Datos del Trabajo", and "Referencias Personal y Familiar". The "Datos Personales" tab is active. The form contains the following fields: "Estado" (set to "#[...estado.inputValue]"), "Nro. Solicitud" (set to "#[...Solicitud.inputValue]"), "Cedula" (set to "#[...cedula.inputValue]"), "Codigo" (set to "#[...CI.Codigo.inputValue]"), "Nombres" (set to "#[...CI.Nombres.inputValue]"), "Sexo" (radio buttons for "Masculino" and "Femenino"), "Estado Civil" (set to "#[...CI.EstadoCivil.inputValue]"), "Cargas Familiares" (set to "#[...CI.CargasFam.inputValue]"), "Direccion" (set to "#[...CI.Direccion.inputValue]"), "Telefono" (set to "#[...CI.Telefono.inputValue]"), "Provincia" (set to "#[...CI.Provincia.inputValue]"), "Cuidad" (set to "#[...CI.Cuidad.inputValue]"), "Sector" (set to "#[...CI.Sector.inputValue]"), "Domicilio" (set to "#[...CI.Domicilio.inputValue]"), "Telf. Domicilio" (set to "#[...CI.TelfDomicilio.inputValue]"), "CI Conyuge" (set to "#[...CI.CiConyuge.inputValue]"), and "Nom. Conyuge" (set to "#[...CI.NomConyuge.inputValue]"). A "Registro Civil" button is located below the "Cedula" field.

Figura 5.26 Ventana de Aprobación o Rechazo de una Solicitud de Crédito

En la siguiente figura se verifica que la solicitud aprobada se haya registrado en la base de datos.

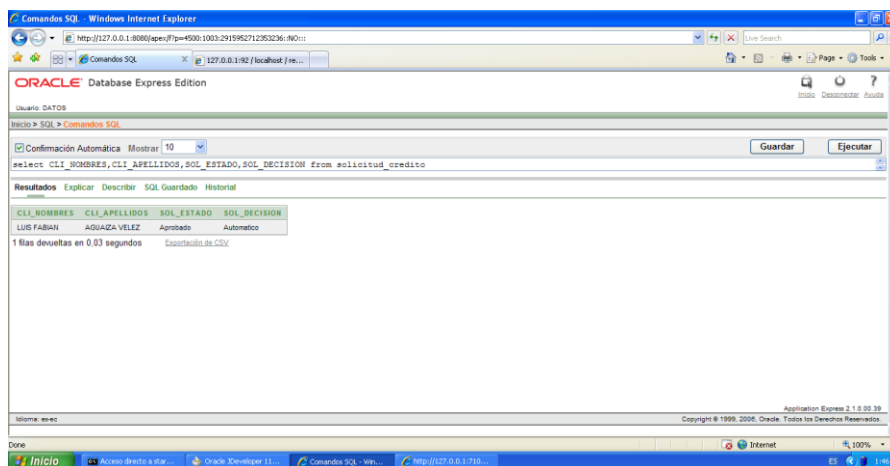


Figura 5.27 Registro de una solicitud Aprobada

Se concluye que todas las herramientas deben estar bien configuradas, ya que si se omite algún paso que se vio anteriormente, puede tener algún problema con algún adaptador de datos, haciendo que los *web services* no identifiquen las conexiones de los adaptadores o del pool de conexiones.

## CAPITULO 6

### CONCLUSIONES Y RECOMENDACIONES

#### 6.1 Conclusiones

- Oracle *BPEL Process Manager* proporciona una metodología y un ambiente de trabajo para lograr la interoperabilidad entre aplicaciones, las cuales pueden estar desarrolladas en diferentes lenguajes y ejecutadas en distintas plataformas. Esto ofrece innumerables beneficios para las empresas de TI, mejorando los tiempos de respuesta y la flexibilidad en la ejecución de la lógica de negocios.
- La tecnología de orquestación *Oracle BPEL Process Manager* ofrece un ambiente comprensivo, robusto y dinámico, ya que emplea los estándares *web services*, adaptadores JCA y protocolos nativos. Además, presenta un diseñador gráfico de procesos que facilita el desarrollo de orquestaciones multiplataforma, dando un gran alcance en la integración de los procesos de negocios de las empresas.
- La orquestación de servicios web que se ha creado en el presente proyecto a través de *Oracle BPEL Process Manager*, tiene un alto nivel de flexibilidad, ya que mediante la aplicación creada se puede demostrar que Oracle BPM es una solución a las necesidades de integración de aplicaciones de negocios, porque se pueden automatizar la lógica de negocios y al mismo tiempo se pueden tomar decisiones en tiempo real que darán solución a los requerimientos del negocio.
- La integración de aplicaciones multilenguaje y multiplataforma que se orquestaron fueron analizadas, diseñadas e implementadas aplicando los conocimientos obtenidos en la carrera universitaria, los cuales facilitaron el análisis detallado de los procesos de negocio que se debían implementar.

- Las tecnologías de Oracle SOA Suite son fundamentales a la hora de diseñar, codificar, mapear, desplegar, validar y monitorear las distintas aplicaciones creadas en el presente proyecto. El IDE JDeveloper junto con el Framework Oracle ADF, proporcionan las herramientas necesarias para trabajar de forma dinámica y flexible a la hora de realizar la orquestación de servicios web y al momento de diseñar una página web para invocar estos servicios.
- Se puede concluir entonces que *Oracle Bpel Process Manager* en conjunto con las herramientas de *Oracle Soa Suite*, son tecnologías de gran alcance para la integración de aplicaciones multilenguaje y multiplataforma. La flexibilidad que se maneja al momento de orquestar servicios web y la facilidad de monitorear los flujos de trabajo, permiten destacar las ventajas que pueden tener los sistemas integrados a la hora de manejar un flujo notable de datos.

## 6.2 Recomendaciones

- El presente proyecto demuestra que *Oracle BPEL Process Manager* puede mover grandes volúmenes de información, a través de los *web services* que se generan en una orquestación de servicios. Por lo tanto, se recomienda que al momento de realizar una integración de sistemas multilenguaje o multiplataforma, es necesario definir el objetivo que se quiere alcanzar con este tipo de tecnologías, el cual debe ser analizado por un grupo estratégico de la organización.
- Para llevar a cabo una orquestación de servicios independientes, es recomendable primero esquematizar los sistemas que se van a integrar en la orquestación de servicios, y así poder tener un modelo global de cómo va a fluir la información, desde el momento en que se realiza una solicitud, hasta cuando se recibe la respuesta o información requerida por el usuario.

- Se debe tener en cuenta que el rendimiento de procesamiento de información dependerá de la capacidad que tenga el equipo en donde se implementarán y ejecutarán este tipo de tecnologías. Por lo que se recomienda mejor dividir la carga de trabajo en dos equipos, uno para lo que es el servidor aplicaciones (*Weblogic Server*) y otro equipo para lo que es el desarrollo de aplicaciones en *Oracle Soa Suite*. Esto es necesario, ya que permitirá equilibrar el balance de carga de recursos que este tipo de tecnologías requieren, logrando ejecutar múltiples procesos en un corto lapso de tiempo.

## BIBLIOGRAFÍA

**Arsanjani, Borges y Holley. 2004.** *Arquitectura Orientado a Servicios*. 2004.

**Beecher, Virginia. 2011.** Oracle Fusion Middleware. [Online] 06 2011. [Cited: 01 15, 2012.]

[http://download.oracle.com/docs/cd/E21764\\_01/integration.1111/e10224.pdf](http://download.oracle.com/docs/cd/E21764_01/integration.1111/e10224.pdf).

**Chao M Beck, Stephen Kinder. 2011.** Ciclo de vida SOA. [Online] 2011. [Cited: 01 22, 2012.]

[http://www.ibm.com/developerworks/ssa/websphere/library/techarticles/0812\\_beck/0812\\_beck.html](http://www.ibm.com/developerworks/ssa/websphere/library/techarticles/0812_beck/0812_beck.html).

**Dezhgosha, Kamyar. Angara, Swathi. 2005.** *Web Services Designing Small-scale web Application*. 2005.

**Hwang, Kevin. 2011.** Oracle Fusion Middleware Installation Guide for Oracle SOA Suite. [Online] 2011. [Cited: 04 05, 2012.]

[http://docs.oracle.com/cd/E15523\\_01/doc.1111/e13925/install.htm#INSOA139](http://docs.oracle.com/cd/E15523_01/doc.1111/e13925/install.htm#INSOA139).

**Kennedy, Mark. 2007.** Oracle Bpel Process Manager. [Online] 2007. [Cited: 03 20, 2012.] <http://www.oracle.com/accessibility/>.

**Louridas, Panagotis. 2008.** *Orchestating web services Bpel*. 2008.

**Madani, Muhammad Rafi Aamiri. 2010.** Oracle Bpel Process Manager. [Online] 09 08, 2010. [Cited: 02 04, 2012.] <http://rafi-oracle.blogspot.com/2010/09/oracle-bpel-process-manager.html>..

**Matjaz Juric, Marcel Krizevnik. 2010.** *WS-BPEL 2.0 for SOA Composite Application with Oracle SOA Suite 11g*. Birmingham : Pack Publishing, 2010. ISBN.

**May, Bob. 2012.** Oracle Fusion Middleware. [Online] 2012. [Cited: 05 14, 2012.] [http://docs.oracle.com/cd/E25178\\_01/integration.1111/e10231/intgn.htm](http://docs.oracle.com/cd/E25178_01/integration.1111/e10231/intgn.htm).

**Microsoft. 2012.** SQL Server 2008 R2. [Online] 2012. [Cited: 06 12, 2012.] <http://www.elsasoft.org/>.

**Oracle. 2009.** Introducing the discover web services. [Online] 2009. [Cited: 03 23, 2012.] [http://docs.oracle.com/cd/E15523\\_01/bi.1111/e10412/intro.htm#i1028617](http://docs.oracle.com/cd/E15523_01/bi.1111/e10412/intro.htm#i1028617).

—. **2010.** Oracle Bpel Process Manager. [Online] 2010. [Cited: 03 10, 2012.] [http://download.oracle.com/docs/cd/E21764\\_01/integration.1111/e10223/10\\_bpel.htm](http://download.oracle.com/docs/cd/E21764_01/integration.1111/e10223/10_bpel.htm).

- . **2009.** Oracle Data Integrator. [Online] 2009. [Cited: 03 22, 2012.]  
[http://docs.oracle.com/cd/E15523\\_01/doc.1111/e14860/products.htm#FMWLC190](http://docs.oracle.com/cd/E15523_01/doc.1111/e14860/products.htm#FMWLC190).
- . **2010.** Oracle Fusion Middleware. [Online] 2010. [Cited: 04 09, 2012.]  
[http://docs.oracle.com/cd/E15523\\_01/doc.1111/e14775/adapter\\_tech.htm#CEGEJHCE](http://docs.oracle.com/cd/E15523_01/doc.1111/e14775/adapter_tech.htm#CEGEJHCE).
- . **2009.** Oracle Service Bus. [Online] 2009. [Cited: 03 25, 2012.]  
[http://docs.oracle.com/cd/E12839\\_01/integration.1111/e10223/04\\_osb.htm](http://docs.oracle.com/cd/E12839_01/integration.1111/e10223/04_osb.htm).
- . **2009.** Oracle Weblogic Communication Service. [Online] 05 2009. [Cited: 06 13, 2012.]  
[http://docs.oracle.com/cd/E14571\\_01/doc.1111/e13806/ns\\_config.htm#OWLAG36826](http://docs.oracle.com/cd/E14571_01/doc.1111/e13806/ns_config.htm#OWLAG36826).
- . **2011.** Quick Start Guide of Oracle Soa Suite 11g. [Online] 2011. [Cited: 05 04, 2012.]  
<http://www.oracle.com/technetwork/middleware/soasuite/overview/quickstartguide/soasuite11gr1ps4-459545.pdf>.
- . **2009.** Web Services. [Online] 2009. [Cited: 03 22, 2013.]  
[http://docs.oracle.com/cd/E15523\\_01/web.1111/e13759/overview.htm#WSOVR109](http://docs.oracle.com/cd/E15523_01/web.1111/e13759/overview.htm#WSOVR109).
- Oracle, BPM. 2004.** Oracle Bpel Process Manager. [Online] 2004. [Cited: 03 10, 2012.] [http://download.oracle.com/docs/cd/B25006\\_02/tour/qt/integration003.htm](http://download.oracle.com/docs/cd/B25006_02/tour/qt/integration003.htm).
- PHP. 2012.** PHP. [Online] 2012. <http://php.net/docs.php>.
- Ralph Gordon, Walter Egan. 2010.** Oracle Fusion Middleware Fusion Developer. [Online] 04 2010. [Cited: 12 15, 2012.]  
[http://docs.oracle.com/cd/E14571\\_01/web.1111/b31974/bcintro.htm#ADFFD124](http://docs.oracle.com/cd/E14571_01/web.1111/b31974/bcintro.htm#ADFFD124).
- Tucker A, Sperncer. 1976.** *El sistema del equilibrio financiero*. s.l. : Herrera Hermanos Sucs S.A, 1976.
- Van Horne, James C. 1995.** *Administracion Financiera*. s.l. : Prenstice Hall, 1995.
- Vasudevan, Sheela. 2006.** Oracle Application Server. [Online] 2006. [Cited: 04 11, 2012.] [http://docs.oracle.com/cd/E11036\\_01/integrate.1013/b31005.pdf](http://docs.oracle.com/cd/E11036_01/integrate.1013/b31005.pdf).
- W3C. 2004.** Tecnologías web services. [Online] 2004. [Cited: 03 22, 2012.]  
<http://www.w3.org/TR/2003/WD-ws-arch-20030808/>.
- Walter Egan, Himanshu Marathe. 2011.** Oracle Fusion Middleware Desktop Integration Developer. [Online] 10 2011. [Cited: 06 15, 2012.]  
[http://docs.oracle.com/cd/E15523\\_01/web.1111/e10139/get\\_start\\_dev\\_tools.htm#CIHJEGFA](http://docs.oracle.com/cd/E15523_01/web.1111/e10139/get_start_dev_tools.htm#CIHJEGFA).



