

Universidad del Bío-Bío
Facultad de Ciencias Empresariales
Departamento de Ciencias de la Computación y
Tecnologías de la Información



Integración de la seguridad en XPD/L/BPEL desde Procesos de Negocio Seguro

Tesis para optar al grado de
Magíster en Ciencias de la Computación

Alumno: Gastón Patricio Márquez Ortega
Directores: Dr. Alfonso Rodríguez Ríos
Dr. Eduardo Fernández-Medina Patón

Chillán, Diciembre 2011

Resumen

Los Procesos de Negocio se han convertido en un tema de gran impacto para las organizaciones, debido a que representan la forma en que ellas llevan a cabo sus objetivos estratégicos. Los Procesos de Negocio (Business Process, BP) son un conjunto de actividades relacionadas y estructuradas que producen un determinado servicio o producto para una determinada organización. Por otro lado, la Administración de los Procesos de Negocio (Business Process Management, BPM) considera procesos a los activos estratégicos de una organización que se deben entender y gestionar.

Para representar los Procesos de Negocio existen lenguajes como UML (Unified Modeling Language) y notaciones como BPMN (Business Process Modeling Notation) que permiten describir la funcionalidad del negocio de una empresa. En el último tiempo, BPMN se ha convertido en un estándar de facto para representar los Procesos de Negocio. BPMN es una notación gráfica estandarizada que permite el modelado de Procesos de Negocio en un formato de flujo de trabajo, donde su principal objetivo es que las descripciones de procesos de negocio sean fácilmente legibles y entendibles por todos los participantes del negocio. Es posible encontrar lenguajes más cercanos a la implementación cuyo código puede ser obtenido mediante transformaciones en forma semi-automática desde una especificación de proceso de negocios usando BPMN. XPDL (XML Process Definition Language) es un formato de archivo basado en XML que puede ser usado para intercambiar modelos de Procesos de Negocio entre distintas herramientas. Por otra parte, BPEL (Business Process Execution Language) es un lenguaje utilizado para la composición de Servicios Web, basado en XML y diseñado para el control central de solicitudes de diferentes Servicios Web.

Originalmente la especificación de BPMN no detalla la representación de aspectos de seguridad relacionados con los Procesos de Negocio. Para considerar este aspecto existen propuestas tales como los Procesos de Negocio Seguro (Secure Business Process, SBP) en las cuales se abordan los Procesos de Negocio y la seguridad. Sin embargo, estas propuestas no llegan a especificar si es posible traducir las especificaciones de seguridad a lenguajes como XPDL o BPEL. Por lo tanto, en esta tesis de magíster se aborda la traducción de los requerimientos de seguridad definidos en un SBP hacia los lenguajes XPDL/BPEL usando el requerimiento de seguridad Access Control que está definido en un SBP. De esta manera, se desarrollará un procedimiento que permitirá que un SBP que contiene Access Control pueda ser interpretado entre distintas herramientas (XPDL) y que, a su vez, pueda ser traducido hacia un Servicio Web seguro (BPEL).

Abstract

Business Processes have become a topic of great impact for organizations, because they represent the way they carry out their strategic objectives. Business Processes (BP) are a set of structured activities that produce a service or product for a particular organization. On the other hand, Business Process Management (BPM) considered strategic assets an organization must understand and manage.

To represent the Business Process exists languages like UML (Unified Modeling Language) and notations such as BPMN (Business Process Modeling Notation) for describing the functionality of a company's business. In recent times, BPMN has become a de facto standard for representing Business Processes. BPMN is a standardized graphical notation that allows the modeling of Business Processes in a workflow format, where his main objective is that the descriptions of Business Processes has become easily readable and understandable by all participants of the business. Languages can be found closer to the implementation of which code can be obtained by transformations in a semi-automatic from a specification of Business Processes using BPMN. XPDL (XML Process Definition Language) is a file format based on XML that can be used to exchange business process models between different tools. On the other hand, BPEL (Business Process Execution Language) is a language used for composing Web Services, XML-based and designed for central control of different Web services applications.

Originally BPMN specification does not detail the representation of secure issues associated with Business Processes. To consider this aspect there are proposals such as Secure Business Process (SBP) in which addresses the Business Process and security. However, these proposals fail to specify whether it is possible to translate safety specifications languages such as XPDL or BPEL. Therefore, in this thesis deals with the translation of the security requirements defined in a PBS into the languages XPDL/BPEL using the Access Control security requirement that is defined in a SBP. Thus, we will develop a procedure that will allow a SBP containing Access Control can be performed between different tools (XPDL) and, in turn, can be translated into a secure Web Services (BPEL).

Deseo dedicar esta tesis a:

A mis padres, Carmen y Gastón y mi hermano Francisco, ya que ellos han sido un pilar fundamental en mi formación. Su apoyo constante y fortalezas infinitas hicieron que nunca me rindiera, incluso en tiempos de adversidad. Estaré eternamente agradecido.

A Michelle, por su constante apoyo, alegría y sabias palabras. Gracias a ti todo esto fue posible. Eres un ángel.

Muchas gracias a todos.

Agradecimientos

Quisiera dar mis más sinceros agradecimientos al Sr. Alfonso Rodríguez Ríos, profesor guía. Para mí fue un honor haber trabajado nuevamente con usted en esta tesis de magíster bajo su dirección y le estaré siempre agradecido porque ha dedicado su valioso tiempo a ello.

Quisiera también agradecer al programa de Magíster en Ciencias de la Computación por la oportunidad que me ofrecieron y por todos los conocimientos entregados para conocer un poco más el mundo que nos rodea. A los docentes del programa que me entregaron las herramientas y habilidades para poder realizar la tesis. Y a la Universidad del Bío-Bío por el constante apoyo para que esta tesis fuese realizada, mediante la ayuda de becas.

Finalmente a todos a todos aquellos que han contribuido a la realización de este trabajo.

Gastón Márquez

Índice general

Índice general	1
Índice de figuras	3
Índice de tablas	5
1 Introducción	6
1.1. Hipótesis y objetivos	8
1.2. Organización de la tesis de magister	8
2 Conceptos relacionados	10
2.1. Procesos de Negocio	10
2.2. Procesos de Negocio Seguro	11
2.3. Business Process Modeling Notation (BPMN)	13
2.4. XML Process Definition Language (XPDL)	17
2.5. Business Process Execution Language (BPEL)	18
2.5.1. Relación entre XPDL y BPEL	19
2.6. Web Services Description Language (WSDL)	20
3 Estado del arte	21
3.1. Propuestas para integrar la seguridad en XPDL/BPEL desde Procesos de Negocio	22
3.1.1. Incorporando requerimientos de seguridad en la composición de servicios: desde el modelado hacia la ejecución	24
3.1.2. Especificación de requerimientos de seguridad en la administración de los Procesos de Negocio orientado a servicios	25
3.1.3. Traduciendo BPMN hacia BPEL	27
3.1.4. Ingeniería de Procesos de Negocio con acuerdos a nivel de servicios	29
3.2. Resumen	29
4 Integración de la seguridad en XPDL/BPEL	31
4.1. Propuesta de incorporación de la seguridad en XPDL	32
4.2. Propuesta de incorporación de la seguridad en BPEL	34

4.2.1.	Esquema de la integración de la seguridad en BPEL	34
4.2.2.	Propuesta de integración de la seguridad en BPEL desde un SBP	35
4.2.2.1.	Creación del BPMN Seguro (Bloque 1)	37
4.2.2.2.	Creación de BPEL/WSDL desde BPMN Seguro (Bloque 2)	37
4.2.2.3.	Creación del Servicio Web seguro (Bloque 3)	41
4.3.	Resumen	41
5	Ejemplo ilustrativo	43
5.1.	Implementación de la integración de la seguridad en XPDL	45
5.2.	Implementación para integrar la seguridad en BPEL	47
5.2.1.	Creación de los estereotipos de seguridad (Bloque 1)	47
5.2.2.	Obtención del código BPEL y archivos WSDL (Bloque 2)	48
5.2.3.	Creación del Servicio Web seguro (Bloque 3)	53
5.3.	Principales dificultades	55
5.4.	Resumen	57
6	Conclusiones	58
6.1.	Principales aportes	60
6.2.	Contraste de resultados	60
	Bibliografía	61
A	Anexo: Código XPDL del ejemplo ilustrativo	67
B	Anexo: Código BPEL del ejemplo ilustrativo	90

Índice de figuras

2.1. Requerimientos de seguridad y su notación asociada [68]	11
2.2. Metamodelo BPD con los requerimientos de seguridad [68]	11
2.3. Núcleo de elementos de BPMN [60]	14
2.4. Actividades [60]	15
2.5. Eventos [60]	15
2.6. Compuertas [60]	16
2.7. Conexiones [60]	16
2.8. Artefactos (Artifacts) [60]	17
2.9. Carriles (Swimlanes) [60]	17
2.10. Conexión entre BPEL v XPDL	19
3.1. Método de revisión adaptado por [12]	21
3.2. Entorno de la herramienta Sec-MoSC [77]	24
3.3. Ejemplo de seguridad en BPMN [50]	26
3.4. Reducción de un diagrama BPMN en componentes [63]	28
4.1. Rol de XPDL/BPEL en la integración de la seguridad desde un SBP	31
4.2. Código XSD que especifica la integración de los atributos extendidos a XPDL	33
4.3. Código XPDL para la integración de Access Control	33
4.4. Marco general de la solución a la integración de la seguridad desde BPMN seguro a BPEL	34
4.5. Recorrido para integrar la seguridad en BPEL desde un SBP	35
5.1. SBP de entrega de ficha a paciente en un centro de atención de salud	44
5.2. Código XPDL de rellenar la ficha clínica con el requerimiento de seguridad Access Control	45
5.3. Interfaz de TIBCO para describir en detalle las actividades	46
5.4. Interfaz de TIBCO para describir los atributos extendidos	46
5.5. Recorrido de transformación a un Web Service Seguro desde un SBP mostrando los principales elementos utilizados	47
5.6. Estereotipos de seguridad en Eclipse Indiao	48

5.7. Modelo BPEL obtenido del BPMN seguro de Proceso de entrega ficha usando la herramienta BPMN2BPEL	49
5.8. Trozo de código BPEL de la actividad Rellenar ficha clínica donde no se especifica seguridad	50
5.9. Trozo de código BPEL de la actividad Rellenar ficha clínica	50
5.10. Trozo de código BPEL del proceso de préstamo	51
5.11. Trozo de código BPEL donde se detallan las variables que se utilizarán en el Servicio Web	51
5.12. Política de seguridad de Access Control	52
5.13. Código WSDL donde se detalla el ingreso de la seguridad	52
5.14. Código de rampart-client-policy.xml	54
5.15. Fragmento de clase Java para la administración de la creación del Servicio Web seguro	54
5.16. Inicio de Servicio Web seguro entrega ficha	55

Indice de tablas

2.1. Nuevos requerimientos de seguridad y elementos de BPD [68]	12
3.1. Escala de valor global de la seguridad	26
3.2. Elementos de seguridad BPMN	27
3.3. Resumen de los trabajos relacionados	29
3.4. Resumen aporte de artículos	30
4.1. Requerimientos de seguridad y su relación con BPMN-BPD y WS-Security/Policy .	41

Capítulo 1

Introducción

Los Procesos de Negocio se han convertido, en el último tiempo, en un tema de gran importancia para las empresas. Un Proceso de Negocio es un conjunto de actividades y tareas que se relacionan lógicamente con el objetivo de lograr un resultado de negocio definido [60]. Cada Proceso de Negocio posee una entrada, función y salida. Las entradas son pre-requisitos para que una función pueda ser aplicada obteniendo salidas resultantes. El enlace entre Procesos de Negocio y la generación de cadenas de valor, lleva a algunos participantes del negocio a ver los Procesos de Negocio como los flujos de trabajo que efectúan las actividades de una organización. Éstos cuentan con una serie de actividades de valor con el propósito de lograr un objetivo empresarial común. Estos procesos son importantes para cualquier organización, ya que pueden generar ingresos y, a menudo, representan una proporción significativa de los costos [52].

Como un enfoque gerencial, la Administración de los Procesos de Negocio (Business Process Management, BPM) considera procesos a los activos estratégicos de una organización que se deben entender, gestionar y mejorar para ofrecer productos y servicios de calidad a los clientes [26]. BPM va un paso más allá al afirmar que este enfoque puede ser apoyado y sustentado a través de la tecnología para garantizar la viabilidad del enfoque de gestión en momentos de crisis y cambio. De hecho, BPM es un enfoque para integrar en una organización la capacidad de cambio, es decir, cambios tanto en el capital humano como tecnológico. Además, es considerado un componente crítico de la cooperación operativa de la inteligencia de las empresas para ofrecer soluciones en tiempo real e información procesable [29].

Por otro lado, las organizaciones de hoy requieren garantizar la integridad, confidencialidad, disponibilidad y privacidad de la información que manejan y procesan, siendo las Tecnologías de Información y la Arquitectura de Seguridad de la Información (Information Security Architecture, ISA) herramientas que ayudan a garantizar la seguridad [16]. ISA permite identificar los elementos y los componentes necesarios para definir, normar, implantar, monitorear y auditar los requerimientos de seguridad con una visión de negocios apoyada en tres factores críticos de éxito: Recursos Humanos, Procesos de Negocio y Tecnología. ISA, por otro lado, permite abordar los cambios que ocurren en las organizaciones, ya que en éstas todo cambia con el paso del tiempo, como por ejemplo las personas, la tecnología, la forma de hacer negocio, los procesos, los volú-

menes de información, los riesgos, entre otros. Así mismo, la seguridad también varía, por lo que hace extremadamente complejo determinar el nivel de seguridad requerido para tratar de mitigar estos nuevos riesgos [16].

Una parte importante de los Procesos de Negocio es la representación o la forma en que ellos pueden ser descritos, para ello existen notaciones como UML (Unified Modeling Language) [81] y BPMN (Business Process Modeling Notation) [60] que representan la funcionalidad del negocio donde en el último tiempo, BPMN se ha convertido en la notación más usada para representar los Procesos de Negocio [72]. BPMN es una notación gráfica estandarizada que permite el modelado de Procesos de Negocio en un formato de flujo de trabajo. El principal objetivo de BPMN es proveer una notación estándar que sea fácilmente legible y entendible por todos los participantes del negocio. Entre estos participantes están los analistas de negocio, los desarrolladores y los gerentes y administradores del negocio [60]. Por otro lado, la representación de los Procesos de Negocio descritos en BPMN pueden ser traducidos en lenguajes de programación con el objetivo de llevar la lógica del negocio entre distintas herramientas de desarrollo y a Servicios Web. Entre ellos se pueden distinguir XPDL y BPEL [66].

El primero de ellos, XPDL (XML Process Definition Language), es un formato de lenguaje basado en XML que puede ser usado para intercambiar modelos de Procesos de Negocio entre distintas herramientas [15], tiene un concepto de líneas que señalan el camino a seguir [79]. XPDL contiene extensiones que le permiten representar todos los aspectos de BPMN y, a su vez, proporciona una notación gráfica para facilitar la comunicación entre usuarios de negocio [15] transformando esa notación en un archivo XML con el objetivo de que se pueda intercambiar entre distintas herramientas para que los distintos actores del negocio puedan ver el diagrama BPMN. Por otro lado, BPEL (Business Process Execution Language) es un lenguaje estandarizado por OASIS para la composición de Servicios Web. Es un lenguaje basado en XML que ha sido diseñado para el control central de solicitudes de diferentes Servicios Web con cierta lógica de negocio que ayuda a la programación a gran escala [78]. BPEL cumple un rol de orquestador cuya funcionalidad es ordenar y determinar la ejecución de distintos Servicios Web [24]. A través de un documento XML, un analista de negocio es capaz de representar la lógica y los elementos con los que se verá relacionado un Proceso de Negocio, donde estos elementos serán los Servicios Web y la lógica del negocio. Del mismo modo, BPEL es un lenguaje de alto nivel que lleva el concepto de servicio un paso más adelante al proporcionar métodos de definición y soporte para flujos de trabajo y Procesos de Negocio [2]. El enfoque sobre Procesos de Negocios modernos más el complemento de los lenguajes WSDL (Web Services Description Language) y XLANG (extensión de WSDL), guiaron a BPEL a adoptar los Servicios Web como su mecanismo de comunicación externa. Así, las facilidades de mensajería BPEL dependen del uso del WSDL para describir los mensajes que entran y salen [31].

En los párrafos anteriores, se ha mencionado que en las organizaciones existen Procesos de Negocio y que éstos se pueden gestionar (BPM), además, se ha descrito que el poseer algún tipo de arquitectura de seguridad es una estrategia que determina el éxito o fracaso de una empresa

Por otro lado, se ha indicado que los Procesos de Negocio se pueden representar mediante la notación BPMN y que, a su vez, esta notación puede ser traducida en lenguajes como XPD L y BPEL. No obstante, en la especificación original de BPMN no está detallada la representación de aspectos de seguridad en relación con los Procesos de Negocio. Es por eso que existen autores como Rodríguez et al. [68] que han trabajado con los Procesos de Negocio y la seguridad, llegando a proponer Procesos de Negocio Seguro (Secure Business Process, SBP) en que se ha considerado el punto de vista del analista de negocio en relación con la seguridad. Esta propuesta define un metamodelo para BPMN en el cual se agregan requerimientos de seguridad, los cuales tienen un rol particular dentro del diagrama BPMN. Pero esta propuesta no especifica si existe alguna traducción hacia los lenguajes XPD L/BPEL desde un SBP. Por lo tanto, esta tesis de magister apunta a la traducción de los requerimientos de seguridad definidos en un SBP hacia XPD L/BPEL, usando el requerimiento de seguridad Access Control que está definido en un SBP. De esta manera, se desarrollará un procedimiento que permitirá que un Proceso de Negocio Seguro que contiene Access Control pueda ser interpretado entre distintas herramientas (XPD L) y que, a su vez, pueda ser traducido hacia un Servicio Web seguro (BPEL), desde el punto de vista del analista de negocio.

1.1. Hipótesis y objetivos

El objetivo general de esta tesis de magister queda expresado por:

Integrar requerimientos de seguridad en XPD L/BPEL a partir de Procesos de Negocio Seguro

En función de este objetivo general, se pueden plantear los siguientes objetivos específicos.

1. Integrar la seguridad en XPD L con el objetivo de que, desde un diagrama BPMN seguro, se pueda traducir a código XPD L abarcando requerimientos de seguridad.
2. Integrar la seguridad en BPEL desde un diagrama BPMN seguro, con el objetivo de crear Servicios Web seguros.

Por lo tanto, la principal hipótesis de esta tesis es:

Es posible integrar mecanismos de seguridad en los lenguajes XPD L/BPEL desde Procesos de Negocio Seguro

1.2. Organización de la tesis de magister

Esta tesis de magister se organiza de la siguiente manera:

Capítulo 2. Conceptos relacionados. En este capítulo se presentan las definiciones y elementos que se relacionan con la tesis.

Capítulo 3. Estado del arte. En el capítulo tres se muestra un detalle de los principales trabajos que se relacionan con la integración de la seguridad en XPDL/BPEL desde Procesos de Negocio Seguro, haciendo una distinción entre aquellos trabajos diseñados con ese propósito.

Capítulo 4. Integración de la seguridad en XPDL/BPEL. En este capítulo se tratará el planteamiento de la integración de la seguridad tanto en XPDL como BPEL, proponiendo un camino de integración de manera semi-automática utilizando herramientas tecnológicas.

Capítulo 5. Ejemplo ilustrativo. En este capítulo se desarrollará un ejemplo ilustrativo relacionado con la entrega de ficha a un paciente en donde se aplicarán las propuestas definidas en el capítulo 4.

Capítulo 6. Conclusiones. Por último, en este capítulo se verificará si los objetivos propuestos en esta tesis fueron cumplidos.

Capítulo 2

Conceptos relacionados

En este capítulo se introducirán los conceptos que se asocian con la propuesta de esta tesis. las secciones 2.1 ,2.2 y 2.3 se describirán los Procesos de Negocio, Procesos de Negocio Seguro y la notación BPMN. Luego, en las secciones 2.4, 2.5 y 2.6 se especificarán los lenguajes que interpretan un Proceso de Negocio descrito en BPMN. los cuales son: XPD L. BPEL v WSDL.

2.1. Procesos de Negocio

Un Proceso de Negocio (BP, Business Process) es un conjunto de tareas y/o actividades que se relacionan lógicamente para lograr un resultado de negocio definido [52]. Cada Proceso de Negocio tiene sus entradas, funciones y salidas. Las entradas son requerimientos que se deben tener antes de que una función pueda ser aplicada. Cuando una función es aplicada a las entradas de un método, se obtienen salidas. Davenport [17] que define un BP como “un proceso estructurado y un conjunto de actividades diseñadas, para producir una salida determinada para un cliente o mercado específico. Esto implica un fuerte énfasis en saber cómo se trabaja dentro de una organización... Un proceso es, pues un orden específico de actividades de trabajo a través del tiempo y del espacio, con un principio y un fin, entradas y salidas claramente definidas...”. Por otro lado, Hammer et al. [29] definen un Proceso de Negocio como “una colección de actividades que toma uno o más tipos de entrada y crea una salida que es de valor para el cliente”. Las organizaciones pueden elegir entre muchas metodologías sofisticadas para la captura y el diseño de modelos para adecuarse a su propósito.

Por otro lado, se llama Business Process Management (BPM) a la metodología empresarial cuyo objetivo es mejorar la eficiencia a través de la gestión sistemática de los Proceso de Negocio mediante el modelado, automatización, integración, monitoreo y optimización de forma continua [31].

2.2. Procesos de Negocio Seguro

El Proceso de Negocio Seguro (Secure Business Process, SBP) es un concepto introducido por Rodríguez et al. [68] que se basa en una extensión al diagrama BPD (Business Process Diagram) [60] con estereotipos de seguridad. Para visualizar la propuesta de los Procesos de Negocio Seguro se mostrará el metamodelo con los requerimientos de seguridad en la Figura 2.1.

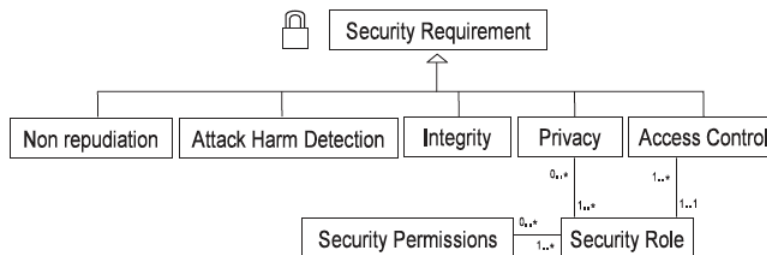


Figura 2.1: Requerimientos de seguridad v su notación asociada [68]

La extensión del metamodelo de BPD, usando el metamodelo de requerimientos de seguridad, se puede ver en la Figura 2.2. La idea principal es crear una clase Business Process Diagram que contenga los estereotipos de seguridad. Esta nueva clase se llama Secure Business Process Diagram.

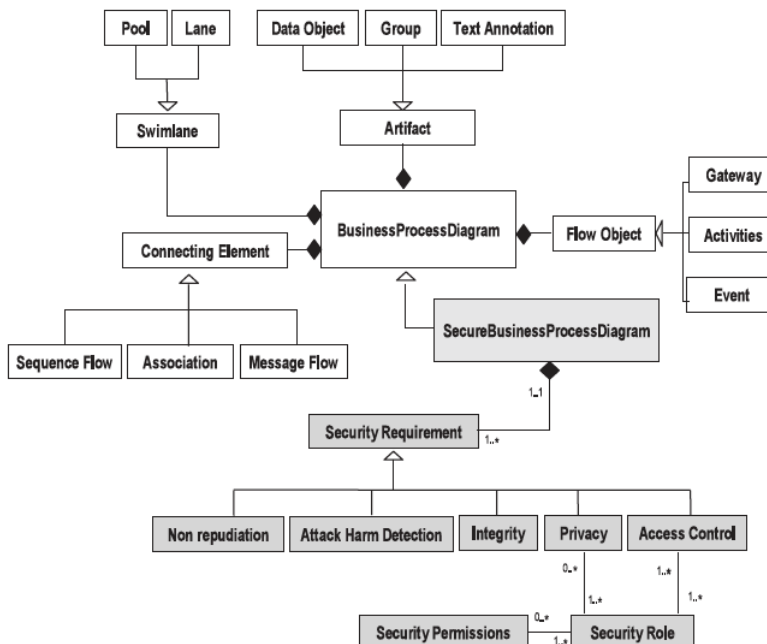


Figura 2.2: Metamodelo BPD con los requerimientos de seguridad [68]





Todos los estereotipos de seguridad tienen relación con algún elemento del diagrama BPD.

A continuación, se mostrará en la Tabla 2.1 los requerimientos de seguridad y su relación con BPD, esto quiere decir que, por ejemplo, Access Control sólo se podrá especificar en Pool, Lanes, Grupos y Actividades, pero no en Flujos de mensajes y Objetos de datos.


	Pool	Lane	Grupo	Actividad	Flujo de mensaje	Objeto de datos
Non Repudation					◆	
Attack Harm Detection	◆	◆	◆	◆	◆	◆
Integrity					◆	◆
Privacy	◆	◆	◆			
Access Control	◆	◆	◆	◆		
Security Roles	◆	◆	◆			
Security Permissions				◆	◆	◆

Tabla 2.1: Relación de los requerimientos de seguridad con los elementos de BPD [60]

Una descripción más detallada de los requerimientos de seguridad, incluyendo su representación gráfica se muestra a continuación.

- **Non Repudation** : establece la necesidad de evitar la denegación de cualquier aspecto de la interacción del diagrama. Desde la perspectiva del analista de negocios, No repudio representa la necesidad de proteger una determinada interacción, de manera que minimice los potenciales problemas. Desde la perspectiva de la seguridad, esta especificación implica la generación de roles de seguridad.
- **Attack Harm Detection** : se define como la detección, registro y notificación de una acción de ataque y amenaza, ya sea que tenga éxito o fracase. Desde la perspectiva del analista de negocios, este requisito representa una señal de atención sobre los elementos en que se indica. Desde el punto de vista de la seguridad, esta especificación implica mantener un registro de los eventos (ataques y amenazas) ocurridos sobre elementos potencialmente vulnerables.
- **Integrity** : establece el grado de protección de una corrupción. Esto quiere decir que cualquier mensaje tiene que estar protegido de los ataques mal intencionados y no autorizados. La integridad se especifica en los datos del objeto y flujos de mensajes.
- **Privacy** : está relacionada con condiciones de protección de la información acerca de un determinado individuo o entidad, limitando el acceso a partes no autorizadas para obtener información sensible. Desde el punto de vista del analista de negocios, la especificación de privacidad implica la no revelación (confidencialidad) y no almacenaje (anonimato) de la información acerca de un determinado rol. Desde el punto de vista de la seguridad, la

especificación de privacidad con confidencialidad implica proteger la información acerca de un rol para que no sea develada a terceros.

- **Access Control** : corresponde a la limitación de acceso a recursos sólo a usuarios autorizados. La especificación de este requisito por parte del analista de negocios implica la limitación de acceso a un conjunto de recursos que son valorados como importantes de ser protegidos de manera especial. Desde la perspectiva de la seguridad, esta especificación supone la definición de roles que pueden ser asignados a personas, entidades, programas, dispositivos u otros sistemas y la definición de permisos para acceder a los objetos que se encuentran en el ámbito de la especificación de control de acceso.
- **Security Roles**: contiene la especificación de un rol de seguridad. Se relaciona con todos los requisitos de seguridad y con el registro de auditoría.
- **Security Permissions**: contiene las especificaciones de permisos relacionadas con especificaciones de control de acceso. Un permiso debe contener el nombre del objeto y las operaciones permitidas.

Con estos estereotipos de seguridad, el analista de negocios estará habilitado para expresar requerimientos de seguridad desde su propio punto de vista.

2.3. Business Process Modeling Notation (BPMN)

BPMN (Business Process Modeling Notation) es una notación gráfica estandarizada que permite el modelado de Procesos de Negocio [60]. BPMN proporciona una forma estándar de representar los Procesos de Negocio tanto para propósitos descriptivos de alto nivel, como para detallados y rigurosos entornos de software orientados a procesos [41]. Existen otros lenguajes como UML [81] con el cual también es posible representar Procesos de Negocio.

El objetivo principal de BPMN es crear mecanismos simples y entendibles para representar el negocio de una empresa, y paralelamente, entender su complejidad. Es por eso, que el enfoque adoptado por las empresas ha sido usar elementos gráficos para mostrar a todos los usuarios cuál es la lógica del negocio en la organización [39]. Esto proporciona un pequeño grupo de categorías de notación para que el usuario pueda reconocer fácilmente los tipos básicos de elementos y entender el diagrama. Las cuatro categorías básicas de elementos son [60]:

1. **Flujos de objeto**: son los principales elementos para representar el Proceso de Negocio. Existen tres:
 - a) Eventos
 - b) Actividades
 - c) Compuertas

2. Objetos de conexión: son mecanismo que unen elementos en el diagrama BPMN. Existen tres maneras de conectar los flujos de información:
 - a) Flujos de secuencia
 - b) Flujos de mensajes
 - c) Asociaciones
3. Carriles: almacenan el proceso que describe BPMN. Existen dos grupos primarios:
 - a) Pools
 - b) Lanes
4. Artefactos: son usados para proporcionar información adicional del proceso. Existen tres tipos de artefactos:
 - a) Objetos de datos
 - b) Anotaciones de texto
 - c) Grupo

En la Figura 2.3 se presentan las categorías básicas que conforma el diagrama BPMN, descritas anteriormente.

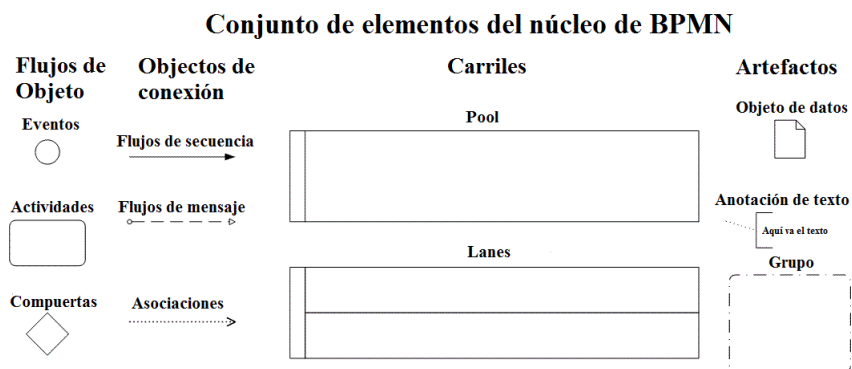


Figura 2.3: Núcleo de elementos de BPMN [60]

En la Figura 2.3 se detalló las cuatro categorías que posee el diagrama BPMN. Cada categoría tiene elementos que, a su vez, se dividen en sub-categorías. A continuación, se describirán las sub-categorías más importantes del diagrama BPMN.

Las actividades son rectángulos con sus vértices redondeados y describen el tipo de actividad que será realizada en el Proceso de Negocio. A su vez, las actividades se pueden representar como: sub-procesos, instancias múltiples, interacciones y compensaciones (ver Figura 2.4).

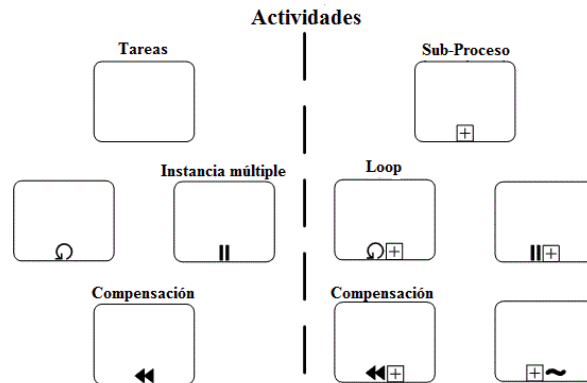


Figura 2.4: Actividades [60]

Los eventos son representados gráficamente por un círculo y describen algo que sucede. Entre ellos, los principales son:

- Evento inicial : Actúa como un iniciador de un proceso. Se representa gráficamente por un círculo de línea delgada.
- Evento intermedio : Indica que algo sucede entre el evento inicial y el evento final. Está representado gráficamente por un círculo de doble línea simple.
- Evento final : Indica el final de un proceso. Está representado gráficamente por un círculo de línea gruesa.

Los eventos, a su vez, poseen distintos tipos, entre los cuales están: mensajes, tiempo, error, cancelar, compensación, regla, enlace, término y múltiple (ver Figura 2.5).



Figura 2.5: Eventos [60]

Las compuertas son representadas por una figura de diamante y determina si se bifurca o se combina los enlaces del proceso dependiendo de las condiciones expresadas por el desarrollador. Existen cuatro tipos de compuertas: XOR. OR. complejas v AND (ver Figura 2.6).

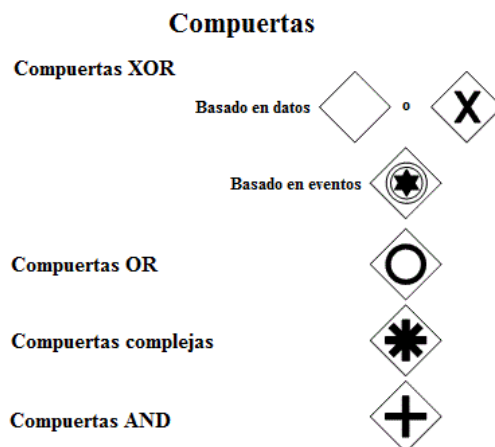


Figura 2.6: Compuertas [60]

Los conexiones permiten unir cada uno de los objetos, entre los cuales se distinguen tres tipos: secuencias, mensaies v asociaciones.

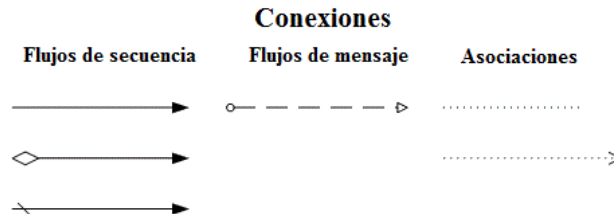


Figura 2.7: Conexiones [60]

Los artefactos permiten a los desarrolladores llevar algo más de información en el modelo o diagrama. De esta manera, el modelo o diagrama se hace más legible. Existen tres tipos de artefactos: objeto de datos, anotación de texto v grupo.

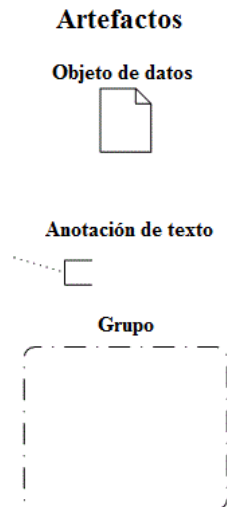


Figura 2.8: Artefactos (Artifacts) [60]

Finalmente, los carriles son un mecanismo de actividades organizadas y categorizadas, basados en organigramas funcionales cruzados y en BPMN se consideran dos tipos: Pool y Lanes.

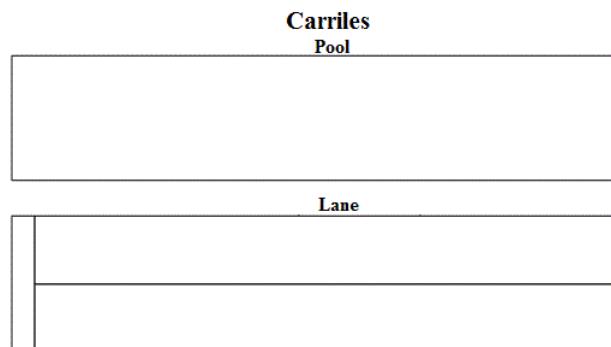


Figura 2.9: Carriles (Swimlanes) [60]

2.4. XML Process Definition Language (XPDL)

XPDL es un lenguaje para la definición de un flujo de trabajo. La definición del proceso consiste en una red de actividades y sus relaciones con los criterios para: indicar el inicio y el término del proceso e información sobre la actividades individuales [87]. Esta interfaz también define una separación formal entre el desarrollo y el entorno, permitiendo que los Procesos de Negocio, generados por una herramienta de desarrollo, puedan ser usados como entrada para un número de diferentes productos [79].

Los estudios realizados por Shapiro [72] y White [88] reflejan la importancia de la adaptación del lenguaje XPDL, ya que las nuevas versiones de BPMN involucran más propiedades que deben ser interpretadas. Shapiro comenta que XPDL 2.2 provee un enfoque de estándar gráfico

hacia la definición de procesos en el diagrama BPMN, esto significa que el archivo XPDL puede ser transformado a otros lenguajes. Además, otra de las ventajas de XPDL 2.2 es que posee una mayor portabilidad en función de la complejidad del modelo. A su vez, White realiza un trabajo experimental de casos de estudios basado en la notación BPMN y su correspondiente en XPDL. A pesar de que existen trabajos [71] [30] que abordan el punto de vista de aplicación relacionados con el lenguaje XPDL, el tema de seguridad es escasamente tratado en la literatura.

Una definición de proceso es capaz de ser interpretada por diferentes productos y ser manejada por una variedad de diferentes herramientas a través del lenguaje XPDL. El trabajo de van der Aalst [1] hace un estudio completo sobre el lenguaje XDPL, donde la propuesta del autor es la definición de 20 patrones de workflow y cómo estos patrones se pueden extender para que puedan ser capturados por XPDL.

2.5. Business Process Execution Language (BPEL)

BPEL (Business Process Execution Language, también llamado WS-BPEL) es un entorno de trabajo basado en XML que permite definir a las empresas los procesos que están conectados dentro o fuera de la organización a través de los Servicios Web [78]. BPEL se convierte en el organizador para unir los Servicios Web en una solución coherente, facilitando su interacción dentro y fuera de una empresa. Una organización que utiliza BPEL se puede constituir de múltiples Servicios Web, creando una aplicación de negocio completamente nueva con una interfaz pública para los usuarios finales. BPEL proporciona una gramática basada en XML para la descripción lógica con el objetivo de controlar y coordinar los Servicios Web que participan en el flujo del proceso. A su vez, esta gramática puede ser interpretada y ejecutada por un motor BPEL, Apache ODE [6], que es controlado (por una parte) por los participantes del negocio. El motor coordina todas las partes del proceso y controla las actividades correctivas cuando se producen excepciones. BPEL utiliza varias especificaciones XML: Web Services Description Language 1.1 (WSDL) [83], esquemas XML 1.0 [82], XPath 1.0 [84] y Extensible Stylesheet Language Transformations (XSLT) 1.0 [85]. Los mensajes WSDL y los esquemas XML proveen el modelo de datos por BPEL, mientras que XPath y XSLT proveen un soporte para la manipulación de datos.

Los conceptos básicos de BPEL pueden ser interpretados de dos formas distintas, de manera abstracta o ejecutable [78] [63]. Un proceso abstracto de BPEL es una especificación parcial del proceso que no intenta ser ejecutada y debe ser declarada como abstracta, mientras que el proceso ejecutable está completamente especificado y debe ser ejecutado. El lenguaje BPEL efectivamente define un formato de ejecución portable para el Proceso de Negocio basado exclusivamente XML. La continuidad del modelo básico entre procesos abstractos y ejecutables en BPEL hace posible que se puedan exportar e importar aspectos públicos que se obtienen en el flujo del Proceso de Negocio.

2.5.1. Relación entre XPD L y BPEL

La comparación entre BPEL y XPD L también ha sido tratada en la literatura existiendo trabajos [66] [71] [13] en donde se discute tal comparación, ya que ambos lenguajes se complementan al momento de traducir un Proceso de Negocio en un Servicio Web. Un diagrama BPMN puede ser traducido a XPD L, éste a su vez, puede ser interpretado por varias herramientas de desarrollo, pero al momento de crear un Servicio Web, XPD L debe ser traducido a BPEL en la misma herramienta en donde se está trabajando. Lo anterior se ve reflejado en la Figura 2.10.

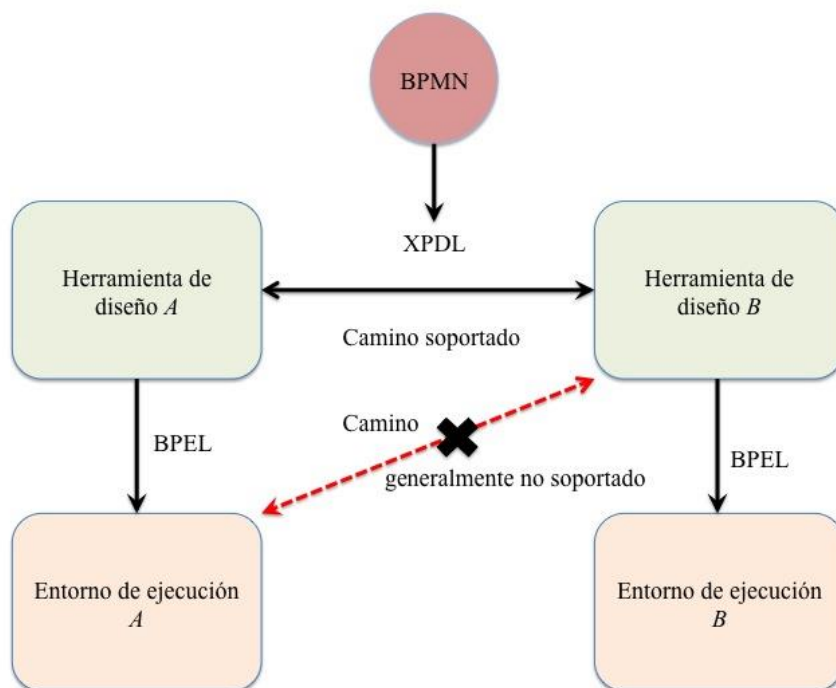


Figura 2.10: Conexión entre BPEL y XPD L

En la parte superior, se muestran las herramientas de diseño A y B. En la parte inferior, son entornos de ejecución correspondientes a las herramientas de diseño. La relación que existe entre estos lenguajes es que XPD L cumple el rol de transportar el diagrama BPMN entre distintas herramientas, pero BPEL interpreta el diagrama BPMN sólo en la herramienta de diseño en la cual interpreta a XPD L, esto quiere decir que si un analista de negocio realiza el diagrama BPMN en la herramienta de diseño A, BPEL ejecutará el diagrama BPMN en el motor de ejecución A. Pero si el analista desarrolla el diagrama BPMN en la herramienta de diseño B, BPEL no podrá interpretar el diagrama BPMN en el motor de ejecución A. Esto es un problema general en las organizaciones que trabajan con BPEL y XPD L al momento de adaptar las tecnologías utilizadas para la traducción y ejecución de sus diagramas BPMN [66] [13].

En la comunidad científica, existe escasa información sobre la traducción desde un archivo XPD L a BPEL y viceversa. Autores como Yuan et al. [91] proponen una herramienta llamada

WorkFlow Transformation systems between XPDL and BPEL (WFTXB), la cual transforma automáticamente (XPDL a BPEL y BPEL a XPDL) los archivos mencionados anteriormente. La importancia de este trabajo, es que es la primera aproximación de una solución de los paradigmas de programación de lenguajes orientados a bloques estructurados y lenguajes orientados a grafos [91].

2.6. Web Services Description Language (WSDL)

WSDL describe la interfaz pública de los Servicios Web. Está basado en XML y describe la forma de comunicación de los mensajes necesarios para interactuar con todos los Servicios Web. La razón de la importancia de los archivos WSDL son el mecanismo de interpretación del código BPEL, por lo tanto, éstos poseen la información y propiedades contenidas en BPEL.

Como los protocolos de comunicación y formatos de mensajes están estandarizados en la comunidad, cada vez se hace importante crear estándares para que la comunicación entre Servicios Web sea más estandarizada. WSDL aborda esta necesidad mediante la definición de una gramática XML para describir servicios de red como colecciones de puertos finales de comunicación capaces de intercambiar mensajes (en relación con los Servicios Web, un mensaje es una solicitud de éstos) [83]. Un documento WSDL define los servicios como puntos finales de puertos. Esto permite la reutilización de los mensajes y tipos de puertos (colecciones abstractas de operaciones). Un puerto se define por la asociación de una dirección de red con un enlace reutilizable y una colección de puertos definen un servicio. Para hacer esta tarea, un documento WSDL utiliza los siguientes elementos en la definición de los servicios de red [83]:

- **Type**: un contenedor para definiciones de tipos de datos utilizados por algún tipo de sistema.
- **Message**: una definición abstracta y escrita de que los datos sean comunicados.
- **Operation**: una descripción abstracta de una acción admitida por el servicio.
- **Port type**: conjunto abstracto de las operaciones apoyadas por uno o más puntos finales.
- **Binding**: un protocolo concreto y específico de formato de datos para un tipo de puerto determinado.
- **Port**: un único punto final que se define como una combinación de un enlace y una dirección de red.
- **Service**: un conjunto de puntos finales.

La idea principal de WSDL es que su definición puede ser utilizada para generar las estructuras de Servicios Web que son ocupadas por las organizaciones en sus negocios y, al mismo tiempo, permite la comunicación con otras organizaciones que utilizan el mismo procedimiento para generar estructuras de Servicios Web más completas.

Capítulo 3

Estado del arte

En este capítulo se mostrará el estado del arte asociado a la propuesta de esta tesis. Este estado del arte está basado en una revisión de la literatura, la cual se realizó bajo las directrices que propone Kitchenham [40], las cuales son apropiadas para la investigación de Ingeniería de Software y se utilizará la adaptación propuesta en [12] que considera un investigador (y no un grupo como en la propuesta original) que es supervisado en el desarrollo de la revisión. Esta revisión de la literatura se estructura mediante etapas y sub-etapas como se describe en la Figura 2.1.

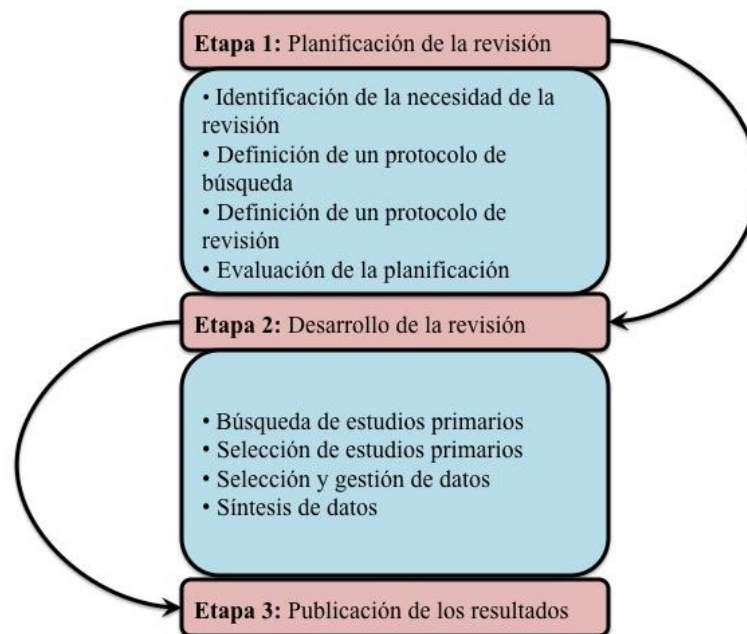


Figura 3.1: Método de revisión adaptado por [12]

La Etapa 1, tiene como propósito específico definir los parámetros más importantes que

serán considerados cuando se realice la revisión. Se debe establecer las razones que justifican llevarla a cabo, la manera en que se hará la búsqueda de trabajos y la forma en que éstos serán revisados, finalmente, se evaluará la planificación realizada. En la Etapa 2, se lleva a cabo la revisión propiamente dicha, su desarrollo está guiado por la planificación de la revisión. Sin embargo, y ya que es un proceso flexible, es posible incluir cambios que mejoren su desempeño. Para complementar la búsqueda y selección de estudios primarios, se ha usado la propuesta de Biolchini et al[11]. Por último, en la Etapa 3 se realizan las publicaciones de los resultados en donde se muestra lo obtenido por la investigación. La propuesta de esta tesis ha sido presentada en congresos nacionales [49] [47] [48], en donde se han mostrado los resultados obtenidos.

Una vez que se ejecuta la revisión de la literatura descrita, en la Figura 3.1, se obtienen trabajos que luego pasarán por una refinación, la cual consiste en la lectura completa del trabajo, con el objetivo de extraer toda la información relevante a la propuesta de esta tesis. En las siguientes secciones se detallarán las propuestas que se acercan al objetivo principal de esta tesis, descrito en la introducción. Por último, se hará un resumen de todos los trabajos seleccionados y tratados en esta revisión.

3.1. Propuestas para integrar la seguridad en XPDL/BPEL desde Procesos de Negocio

Previo a detallar las propuestas de este capítulo, es necesario mencionar que el tema de la seguridad en los Procesos de Negocio ha sido abordado por diversos autores. Esto se debe a que las organizaciones van creciendo y, paralelamente, van creciendo sus vulnerabilidades debido a que aumenta la capacidad del personal, manejan datos más delicados, abarcan nuevos escenarios, entre otras razones. En la literatura es posible encontrar diferentes enfoques en relación a la seguridad en Procesos de Negocio. Algunos autores proponen obtener datos de seguridad en la etapa de obtención de requerimientos [73][74][75], otros proponen una extensión a UML [33][34][37][69] y otros una extensión a BPMN [68][50][89]. Sindre et al. [75] proponen obtener requerimientos de seguridad mediante casos de mal uso, con el objetivo de obtener nuevos requerimientos con mecanismos extra funcionales. Para ello, el autor define dos nuevos conceptos para el diagrama de casos de uso de UML, *misuses cases* (casos de mal uso), *misuser* (actor mal intencionado) y *mal-activity* (actividades mal intencionadas) [73]. De esta manera, se pueden obtener nuevos requerimientos de seguridad asignando funciones que se escapan de lo funcional del sistema [76].

Por el lado de UML, Jürjens et al. [35][33][38] han propuesto diferentes mecanismos y funcionalidades para extender la notación UML con seguridad, resumiendo todo en una propuesta llamada UMLsec [37]. Para ello, se crea un perfil de UML con todos los mecanismos de seguridad integrados al paradigma de orientación de objetos propuesto por UML. UMLsec ha sido probado en varios escenarios con éxito [32][36].

Por el lado de los lenguajes que traducen los Procesos de Negocio, existen propuestas cercanas a la integración de la seguridad en BPEL como es el caso de Charfli et al. [14] que hacen

un estudio de integración de estándares de seguridad en BPEL usando políticas de seguridad con el objetivo de verificar la compatibilidad de la seguridad en la composición del Proceso de Negocio. Siguiendo la misma línea, Rudolph et al. [70] proponen un entorno de trabajo para la ejecución de Servicios Web seguros, a partir de requerimientos de seguridad sobre el tratamiento de los datos en una organización. Propuestas como VxBPEL [42], que son adaptaciones del lenguaje BPEL que capturan variabilidades en los Servicios Web, hacen que BPEL se torne más dinámico. No solo a nivel de Servicios Web se puede tratar BPEL, Juric [31] utiliza WSDL proponiendo una nueva extensión para BPEL, donde se trata de adaptar estos lenguajes a las nuevas exigencias de EDA (Event Driven Architecture). Las políticas de seguridad y BPEL son estudiadas en [44], enfatizando que en Internet no se ha generado mucha preocupación por las políticas de seguridad de las organizaciones, que consumen y recolectan datos personales usando medios automáticos. En este trabajo se propone una transformación gráfica basada en un Proceso de Negocio en BPEL aplicando políticas de seguridad. Ya que los Servicios Web se relacionan con BPEL, López et al. [45] proponen una especificación y diseño sobre servicios de autenticación y autorización avanzados basados en Procesos de Negocio, usando tecnologías y políticas de seguridad tales como: certificados X.509, PKI, PMI, entre otras. BPEL también ha sido estudiado para introducir mecanismos de seguridad heterogéneos en [53], extendiendo la actividad primitiva invoke para que soporte la seguridad. El Control de Acceso es tratado de forma más general en [46] mediante los entornos de colaboración, en donde BPEL se encuentra situado.

En relación con los requerimientos de seguridad definidos en un Proceso de Negocio, Paci et al. [65] y Bertino et al. [9] muestran que extender la seguridad en BPEL dependerá de las tecnologías que soportan a este lenguaje. En el primer trabajo [65] se propone un entorno de trabajo donde realiza controles de acceso usando RBAC (Role-Based Access Control) y en el segundo [9], también propone controles de accesos y autorizaciones usando RBAC con el lenguaje BPCL (Business Process Constraint Language). El control de acceso también es tratado pero de forma más general en [46] mediante los entornos de colaboración, en donde BPEL se encuentra situado. No sólo los Controles de Accesos son una alternativa para BPEL, los protocolos de no repudio [10] también han sido investigados como alternativa de seguridad en BPEL.

A nivel de Servicios Web, Meziane et al. [51] proponen un modelo dinámico de privacidad para los Servicios Web. Este modelo soporta el nivel de negociación de los acuerdos de seguridad con los Servicios Web. A su vez, proponen un protocolo de negociación para tratar las negociaciones de seguridad en un nuevo acuerdo de seguridad. Por otro lado, Gutiérrez et al. [28] proponen PWSec, una aplicación para elicitar y diseñar seguridad en Servicios Web.

En los párrafos anteriores, se ha hablado que la seguridad en los Procesos de Negocio ha sido tratada en distintos niveles: requerimientos, representación del Proceso de Negocio (UML y BPMN), lenguajes que traducen Procesos de Negocio (BPEL) y Servicios Web. En las subsecciones siguientes, se profundizará en aquellos trabajos más cercanos con el propósito de esta tesis

3.1.1. Incorporando requerimientos de seguridad en la composición de servicios: desde el modelado hacia la ejecución

Souza et al. [77] hacen posible la creación de Servicios Web desde Procesos de Negocio con seguridad. A pesar de que este trabajo incorpora la seguridad en los Procesos de Negocio, no posee la formalidad necesaria para definir la seguridad. Se utilizan requerimientos de seguridad clásicos y no funcionales para probar la propuesta. En la Figura 3.2, se puede observar el marco de trabajo llamado Sec-MoSC que es un conjunto de repositorios (para los requerimientos de seguridad), herramientas de entorno de desarrollo y archivos XML.

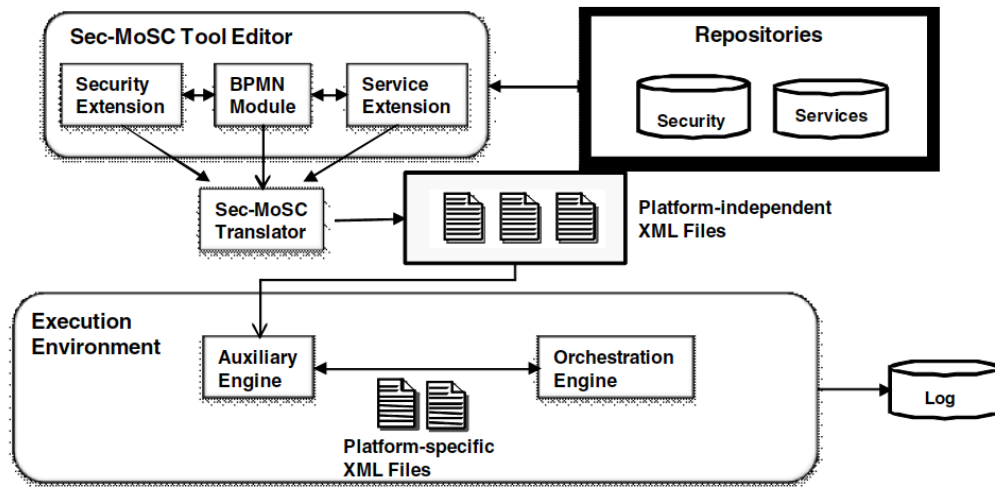


Figura 3.2: Entorno de la herramienta Sec-MoSC [77]

Sec-MoSC proporciona soporte en tiempo de desarrollo y está compuesto por cuatro componentes principales: BPMN, un módulo de extensión de seguridad, un módulo de servicio de extensión y un traductor de Sec-MoSC. El módulo de BPMN permite a los desarrolladores definir un Proceso de Negocio utilizando la notación estándar BPMN. La extensión de seguridad brinda apoyo al modelo de los requisitos de seguridad y se unen a los elementos de BPMN. La extensión de servicio es la responsable de la anotación de la información de servicio (por ejemplo, los servicios de candidato para ejecutar una tarea) en BPMN. Por último, Sec-MoSC se encarga de llevar a cabo la transformación de la notación del modelo BPMN en WS-BPEL y otros archivos de especificación. Los servicios de depósitos y de seguridad almacenan la información sobre los servicios pre-registrados y con el apoyo de los requerimientos de seguridad definidos por los autores. Estos depósitos tienen compatibilidad con las anotaciones del modelo de Procesos de Negocio. El repositorio log posee las propiedades de seguridad de la información de registro generados durante la ejecución. El entorno de ejecución es responsable de la ejecución de la composición de servicios y la composición de los requisitos no funcionales. Esto se lleva a cabo mediante la aplicación de mecanismos a través de la generación de archivos de configuración para el motor de orquestación. Esta generación es realizada por el motor auxiliar, que aplica y administra las con-

figuraciones con el fin de ejecutar la composición de servicios de seguridad. Es importante tener en cuenta que el código BPEL funcional no se altera con el fin de insertar puntos de aplicación. el enfoque que presentan los autores, los puntos de aplicación se especifican en los archivos de configuración generados por el motor auxiliar. Por ejemplo, si el usuario especifica que una comunicación debe ser encriptada a través de un requerimiento de seguridad (por ejemplo, el uso de criptografía), el motor auxiliar recibirá este requisito y se generan configuraciones específicas con el fin de garantizar que los puntos de aplicación se han realizado.

El trabajo presentado anteriormente, especifica requerimientos no funcionales que se ven reflejados en un Proceso de Negocio para luego crear un Servicio Web usando la herramienta Sec-MoSC, pero la herramienta no considera Procesos de Negocio Seguros, no contiene requisitos de seguridad o alguna notación que los especifique, por otro lado, no toma el punto de vista de analista de negocios.

3.1.2. Especificación de requerimientos de seguridad en la administración de los Procesos de Negocio orientado a servicios

La Arquitectura Orientadas a Servicios (Service-oriented Architecture) ofrece una infraestructura flexible que permite que los componentes que son desarrollados de manera independiente, se puedan comunicar de forma perfecta. En el ámbito de flujos de trabajo de la organización, SOA proporciona una base adecuada para ejecutar Procesos de Negocio como una orquestación de múltiples servicios independientes. Junto con el aumento de la conectividad, los correspondientes riesgos de seguridad aumentan exponencialmente [50]. Sin embargo, los requisitos de seguridad generalmente se definen a nivel técnico, a diferencia del nivel organizacional que proporciona un amplio punto de vista sobre los participantes, los activos y sus relaciones en materia de seguridad. En este trabajo, los autores proponen un enfoque para describir la seguridad, requisitos en la capa de Procesos de Negocio y su traducción a la configuración de seguridad concretas para sistemas basados en servicios. Se introducen elementos de seguridad para el modelado de Procesos de Negocio que permiten evaluar la confiabilidad de los participantes sobre la base de los activos de la empresa y la expresión de las intenciones de seguridad, como la confidencialidad y la integridad en un nivel abstracto.

En este trabajo se indica que la organización de la seguridad es el aspecto central de la ingeniería de la seguridad la cual está basada en la gestión de los riesgos que se derivan de las amenazas potenciales a partir de los activos de la empresa. Para evaluar el impacto de las amenazas, los activos deben ser evaluados para determinar su importancia global en una empresa. Esta propuesta se centra en las amenazas relacionadas con el uso de información de identidad y respeto de los derechos asociados (autenticación, autorización, confianza), la información almacenada, transferida y procesada (datos de confidencialidad e integridad de datos) y el funcionamiento del servicio (la integridad del sistema y disponibilidad). La propuesta se resume en la Figura 3.3, donde se describe un Proceso de Negocio en BPMN con elementos de seguridad.

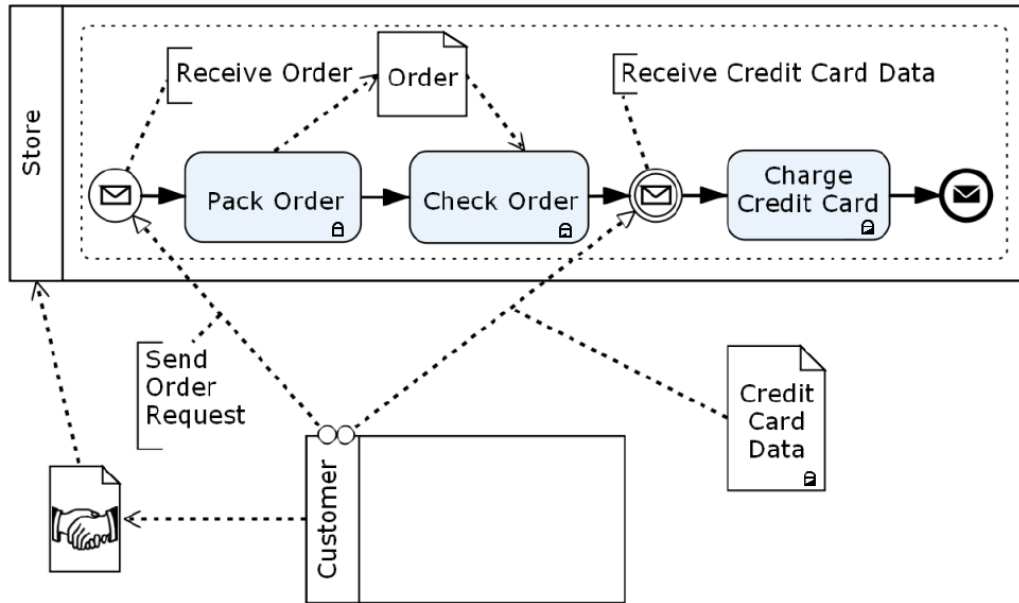


Figura 3.3: Ejemplo de seguridad en BPMN [50]

Los candados que se describen en la Figura 3.3 representan valores de seguridad que están basados en una escala que está detallada en el trabajo (ver Tabla 3.1). Del mismo modo, este trabajo propone los siguiente elementos gráficos para la notación BPMN (ver Tabla 3.2):







	Escala	Descripción
	Extrema	Pone en peligro la vida o la amenaza de la empresa
	Muy alta	Graves consecuencias financieras o de seguridad
	Alta	Impacto en los servicios al cliente y la reputación
	Media	Afecta la misión de la empresa
	Baja	Daño menor financiero y en el impacto de negocios pequeños
	Despreciable	No tiene relevancia en la seguridad

Tabla 3.1: Escala de valor global de la seguridad



	Tipo	Confianza en la organización
	Artefacto	Especifica la relación de confianza entre dos o más participantes
	Artefacto	Especifica las intenciones de la seguridad para un grupo de tareas, los artefactos o pools

Tabla 3.2: Elementos de seguridad BPMN

Este trabajo define la seguridad en los Procesos de Negocio desde un punto de vista de los participantes y activos del negocio, pero no desde el punto de vista del analista del negocio. Por otro lado, representación de la seguridad descrita en la notación BPMN, se ve reflejada por candados que indican el nivel de grado de seguridad y no el punto de vista de requerimientos de seguridad.

3.1.3. Traduciendo BPMN hacia BPEL

En BPMN, los modelos de los Procesos de Negocio son capturados como BPD y éstos a su vez, se componen de elementos que son capaces de representar los aspectos del negocio que se quiere describir. A partir de lo anterior, Ouyang et al. [63] [64] proponen un mecanismo llamado BPMN2BPEL (mejorada por García-Bañuelos [25]) el cual transforma los elementos que están en BPD que representan los Procesos de Negocio en BPEL, usando un algoritmo iterativo que reduce los elementos de BPD a componentes. Esta herramienta está limitada a los siguientes elementos del BPD:

- Eventos iniciales, intermedios v finales.
- Actividad
- Flujo de secuencia
- Compuertas paralelas v XOR

Los otros elementos del BPD no son considerados en este trabajo, lo que limita la representación del Proceso de Negocio. Un componente es un subconjunto de BPD que solamente tiene un punto de entrada y de salida. Pero la definición de un componente, también requiere otra función llamada Fold que reemplaza un componente por una actividad en un BPD. Esta función puede ser utilizada para realizar una reducción iterativa de un BPD hasta que no quede ningún componente en un BPD. La función cumplirá un rol fundamental la cuál reemplaza gradualmente los componente de un BPD a BPEL. A continuación, en la Figura 3.4 se explicará el funcionamiento del algoritmo iterativo y la función Fold.

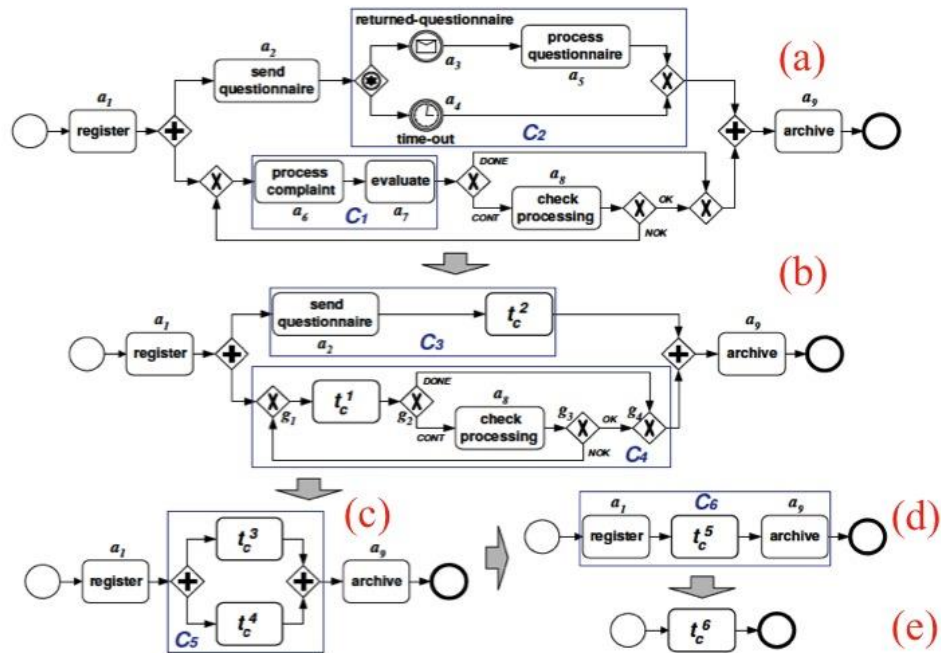


Figura 3.4: Reducción de un diagrama BPMN en componentes [63]

En la Figura 3.4 se presenta un ejemplo donde se describe un Proceso de Negocio en (a). Las actividades son representadas por la letra a_x , donde x representa la cantidad de actividades que hay en el diagrama. Por otro lado, en (a) se puede identificar los componentes que están descritos por la letra C_y , donde y representa la cantidad de componentes que hay en el diagrama. Por lo tanto, en (a) se han identificado nueve actividades y dos componentes. La función Fold esta especificada para reconocer estructuras dentro del diagrama BPMN como se aprecia en C_1 y C_2 . Una vez que se hayan identificado tanto las actividades como los componentes, se aplica el algoritmo iterativo convirtiendo los componentes en actividades que contienen las propiedades de los componentes que fueron reducidos, llegando a la letra (b). Tales actividades se ven reflejadas por la notación t^z_c , donde z representa el número del componente. Luego de aplicar el algoritmo iterativo, se obtienen los diagramas (c), (d) y (e), en que el último diagrama, contiene al diagrama BPMN original reducido en una sola actividad. Por último, al diagrama (e) se le aplica una transformación obteniendo así el correspondiente código BPEL.

A pesar de que el trabajo expuesto aborda la transformación de un diagrama BPMN hacia BPEL, éste no aborda temas de seguridad en su transformación. Por otro lado, dentro de su especificación, tampoco considera extensiones para considerar la seguridad ni en BPEL ni en BPMN.

3.1.4. Ingeniería de Procesos de Negocio con acuerdos a nivel de servicios

Las características de los Servicios Web como la autonomía, la independencia, entre otras, están siendo cada vez más explotadas por las organizaciones. Las empresas no solo exportan sus funciones como Servicios Web, también exportan sus servicios a otras organizaciones para que puedan usarlos. Dado que estos servicios también pueden ser ofrecidos por otros proveedores, no hay algún mecanismo que asegure la confianza y seguridad de sus datos. Es por eso que en el trabajo de Frankova [24] se aborda la ingeniería de los Procesos de Negocio basados en acuerdos a nivel de servicios desde los requerimientos. Este trabajo complementa las metodologías y el desarrollo de Procesos de Negocio bajo en una arquitectura orientada a servicios, enfatizando la seguridad.

Frankova propone una metodología para obtener Servicios Web seguros a partir de acuerdos a nivel de servicios que garanticen la calidad de los requerimientos. Las funciones de seguridad se aplican en BPEL, proponiendo el lenguaje BPEL seguro como mecanismo de especificación de seguridad. Para ello, se desarrolla una extensión al requerimiento WS-Agreement de los Servicios Web, el cual se basa en un algoritmo que proporciona un prototipo que obtiene Servicios Web seguros a partir de acuerdos de nivel de servicio.

A pesar de que esta propuesta aborda el tema de tanto BPEL como Servicio Web seguro, no considera la seguridad en los Procesos de Negocio. Se obtienen Servicios Web seguros desde un enfoque de acuerdos a nivel de servicios, pero no desde el punto de vista del analista de negocio.

3.2. Resumen

En este capítulo se ha descrito el estado del arte de la tesis basado en la revisión sistemática de la literatura en relación con las propuestas de integración de la seguridad en XPDL/BPEL desde Procesos de Negocio. El resumen de los trabajos relacionados se presenta la siguiente Figura 3.3.

Iniciativa	Nº de trabajos	Trabajos
Traducciones BP→XPDL	2	[72] [88]
Traducciones BP→BPEL	2	[63] [64]
Traducciones SBP →XPDL/BPEL	0	–
Traducciones SBP→XPDL/BPEL→Servicio Web seguro	0	–
Traducciones BP con seguridad→XPDL/BPEL→Servicio Web seguro	3	[24] [50][77]
Traducciones XPDL→BPEL	2	[66] [91]
Total	9	

Tabla 3.3: Resumen de los trabajos relacionados

En la Tabla 3.3 se puede observar que existen trabajos que abordan la traducción desde un BP a XPDL y BPEL, además, existen trabajos que traducen un BP con algún tipo de

seguridad (desde el punto de vista de acuerdos a nivel de servicio, grados de impacto, entre otras) a XPD/L/BPEL para obtener un Servicio Web seguro, del mismo modo la traducción entre XPD/L y BPEL también ha sido tratada, pero no existen propuestas que traten la seguridad en XPD/L/BPEL desde un SBP para obtener un Servicio Web seguro.

En la Tabla 3.4 se resume el aporte los trabajos que exclusivamente se relacionan con la propuesta de tesis.

Autor	Propuesta	Iniciativa	Aporte
Frankova	Servicios Web seguros usando acuerdos a nivel de servicios	Metodología	Extensión a WS-Agreement
Menzel et al.	Especificación de requerimientos seguridad	Framework	Seguridad a nivel de Procesos de Negocio
Ouyang et al.	Transformación desde BPMN hacia BPEL	Herramienta	BPMN2BPEL
Souza et al.	Especificación de seguridad Web en Servicios Web a partir de aspectos de requerimientos no funcionales	Framework	Servicios con seguridad desde
definidos en un Proceso de Negocio			Procesos de Negocio

Esta revisión ha permitido mostrar que el tema de integración de especificaciones de seguridad en XPD/L/BPEL desde Procesos de Negocio Seguro es un tema que en la comunidad científica está escasamente tratado, ya que no se ha encontrado alguna propuesta que directamente aborde este tipo de traducciones, lo cual está descrito en la Tabla 3.3. Por lo tanto, esta tesis de magister abordará el vacío que hay en esta área, proponiendo alternativas de integración de la seguridad en los lenguajes XPD/L/BPEL desde SBP para obtener un Servicio Web seguro desde el punto de vista del analista de negocio.

Capítulo 4

Integración de la seguridad en XPDL/BPEL

En este capítulo se describirán las propuestas de integración de la seguridad obtenidas desde un SBP tanto en XPDL como en BPEL. Cada lenguaje cumple un rol particular cuando se habla de traducción de Procesos de Negocio, XPDL cumple la tarea de traspasar la información entre herramientas y BPEL interpreta el Proceso de Negocio en un motor de ejecución para luego crear un Servicio Web. La integración de la seguridad en ambos lenguajes, no cambia los roles descritos anteriormente. En la Figura 4.1, se ilustra la forma en que los SBP serán traducidos a Servicio Web seguro, poniendo énfasis en la seguridad.

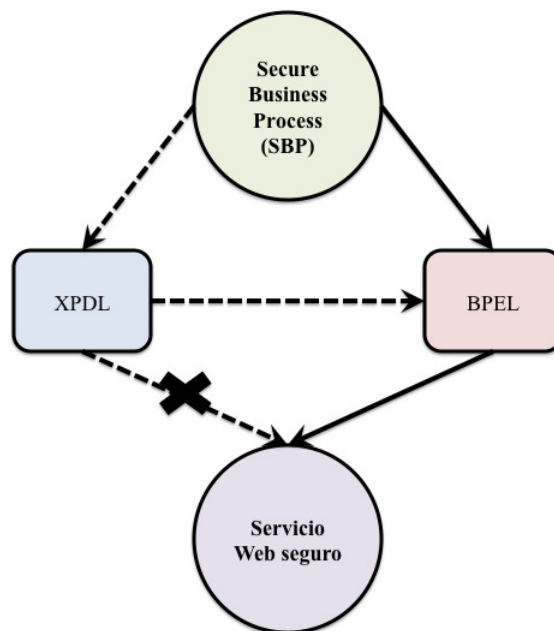


Figura 4.1: Rol de XPDL/BPEL en la integración de la seguridad desde un SBP

La Figura 4.1 muestra que la integración de la seguridad en XPDL/BPEL desde un SBP es posible, pero XPDL no es capaz de generar un Servicio Web seguro ya que la funcionalidad de XPDL es traspasar las propiedades de un diagrama BPMN entre herramientas. Por lo tanto, BPEL puede trabajar de forma independiente para construir un Servicio Web seguro, pero XPDL depende de BPEL para crear un Servicio Web seguro. A pesar de que la seguridad no es considerada en esa propuesta [91], el trabajo da posibilidades de ser extendido para que considere otros mecanismo que complementan a XPDL/BPEL, entre ellos la seguridad. A continuación, en la sección 4.1 se detallará la propuesta de integración de la seguridad en XPDL desde un SBP y en la sección 4.2, se mostrará el camino para integrar la seguridad en BPEL y obtener un Servicio Web seguro a partir del uso simultáneo de tecnologías y otros lenguajes (Web Services Description Language).

4.1. Propuesta de incorporación de la seguridad en XPDL

En el estado del arte se mostró que existe escasa información de integración de la seguridad en XPDL. Este lenguaje no está orientado a extensiones que permitan representar algún mecanismo o notación adicional que pueda dar soporte a extensiones en BPMN. Es por eso, que Shapiro [72] menciona que para enriquecer el diagrama BPMN en XPDL, se debe utilizar elementos ya existentes en el lenguaje. Los atributos extendidos del lenguaje XPDL son propiedades que complementan el lenguaje XPDL y se pueden usar en todas las entidades del diagrama BPMN, estos, a su vez, permiten ampliar la funcionalidad del modelador para satisfacer las necesidades del cliente. El uso de los atributos extendidos puede ser utilizado de dos maneras de acuerdo a lo planteado en la definición de XPDL [79]:

1. Para proporcionar un nombre y un valor para la extensión sin la necesidad de validar la extensión.
2. Para enriquecer más la notación BPMN que se desee realizar.

En la siguiente Figura 4.2 se puede apreciar el código XSD [86] (XML Schema Definition) que permite la integración de los atributos extendidos en el lenguaje XDPL.

```

...
1:</xsd:choice>
2:    <xsd:attribute name="Name" type="xsd:NMTOKEN" use="required"/>
3:    <xsd:attribute name="Value" type="xsd:string"/>
4:  </xsd:complexType>
5:</xsd:element>
6:
7:  <xsd:element name="ExtendedAttributes">
8:    <xsd:complexType>
9:      <xsd:sequence>
10:        <xsd:element ref="xpd1:ExtendedAttribute" minOccurs="0"
11:          maxOccurs="unbounded"/>
12:      </xsd:sequence>
13:    </xsd:complexType>
14:</xsd:element>

```

Figura 4.2: Código XSD que especifica la integración de los atributos extendidos a XPDL

En las líneas 2, 3 y 7 se puede apreciar tanto la definición del nombre y el valor de los atributos extendidos para que puedan ser interpretados por XPDL al momento de intercambiar la definición del proceso como el ingreso de la opción de atributo extendido en el lenguaje XPDL. Este código permite que el lenguaje XPDL soporte extensiones para complementar el diagrama BPMN. Por lo tanto, si se tuviese un SBP con el requerimiento Access Control, la declaración en XPDL quedaría de la siguiente manera:

```

1:<xpd12:ExtendedAttributes>
2:  <xpd12:ExtendedAttribute Name="AccessControl"
3:    Value="...">
4:  </xpd12:ExtendedAttribute>
5:</xpd12:ExtendedAttributes>

```

Figura 4.3: Código XPDL para la integración de Access Control

El código descrito en la Figura 4.3 permite integrar no sólo Access Control (ver líneas de 1 a 5), sino todos los requerimientos de seguridad, ya que como se ha mencionado, XPDL cumple el rol de transportar datos que complementan el diagrama BPMN entre herramientas, pero no es capaz de ser ejecutado en un motor de ejecución. Lo anterior permite a XPDL especificar todos los requerimientos de seguridad que, a su vez, podrían ser interpretados por BPEL y generar un

Servicio Web seguro.

4.2. Propuesta de incorporación de la seguridad en BPEL

En este capítulo, se detallarán los pasos necesarios que permiten la integración de requerimientos de seguridad en BPEL desde un Proceso de Negocio Seguro. La integración de la seguridad en BPEL, conlleva un desarrollo más amplio, ya que el producto final no se transmite entre herramientas (como es el caso de XPD), sino que se ejecuta en forma directa en un motor de ejecución Web. La propuesta que se presentará en esta sección involucra tecnologías que permiten hacer que el camino desde un BPMN seguro hacia un Servicio Web sea posible. Cuando se indica que BPEL es un orquestador [24] [43], es porque une todos los Servicios Web que pueda poseer una empresa bajo un enfoque de negocio. Una de las capacidades limitadas que posee BPEL es el motor de ejecución en donde funciona, en este caso es Apache ODE (más adelante se hablará en detalle del rol de este servidor). lo que también limita la integración de la seguridad.

4.2.1. Esquema de la integración de la seguridad en BPEL

Para lograr la integración de la seguridad desde un BPMN seguro a BPEL, hay que seguir un recorrido que involucra tecnologías que trabajan en conjunto. Cada tecnología, que será descrita en las secciones siguientes, cumplirá un rol con el objetivo de obtener un Servicio Web seguro. El marco general de la solución es descrito en la Figura 4.4.

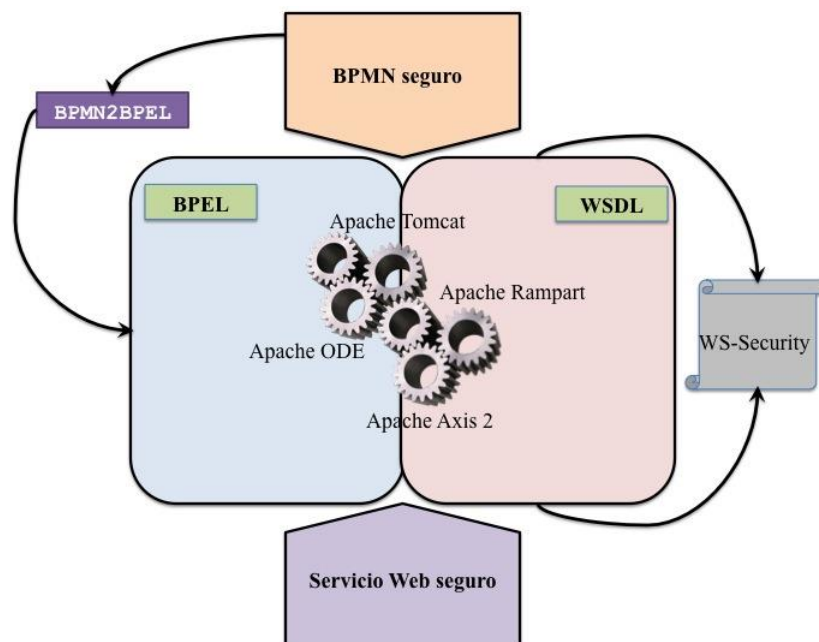


Figura 4.4: Marco general de la solución a la integración de la seguridad desde BPMN seguro a BPEL

La Figura 4.4 identifica todos los componentes de la integración de la seguridad en BPEL, los cuales son: el diagrama BPMN seguro, el código BPEL/WSDL y el Servicio Web seguro. A continuación, se explicará de forma general la propuesta de integración de la seguridad en BPEL y en la subsección 4.2.2, se hablará en detalle de los componentes que realizan esta transformación. En el diagrama BPMN seguro se define la notación gráfica de todos los requerimientos definidos en un SBP. Para efectos de la investigación, en esta tesis de magister se trabajó con el requerimiento de Access Control, el cual es añadido mediante una extensión que será explicada en profundidad en la subsección 4.2.2. Luego, a partir del diagrama BPMN seguro usando la herramienta BPMN2BPEL, es posible obtener el código BPEL. A continuación, se desarrolla el código WSDL que contiene la referencia del requerimiento de seguridad Access Control, mediante la programación de estándares de seguridad para Servicios Web. Para ejecutar estos lenguajes, es necesario que un conjunto de tecnologías desarrolladas por Apache funcionen sincronizadamente, éstas son: ODE, Axis2, Rampart y Tomcat. La selección de estas tecnologías se hizo teniendo en cuenta las referencias citadas en el capítulo 3, ya que, la mayoría de los autores que han tratado con ellas, concuerdan en que son adecuadas para la trabajar con la seguridad tanto a nivel de BPEL como de Servicio Web. Por último, a partir de los componentes anteriores, se obtiene un Servicio Web seguro que tiene el requerimiento de seguridad definido en el diagrama BPMN seguro.

4.2.2. Propuesta de integración de la seguridad en BPEL desde un SBP

En esta subsección, se explicará en mayor profundidad la propuesta de integración de la seguridad en BPEL desde un SBP, resumida en la Figura 4.1 (ver Figura 4.5).

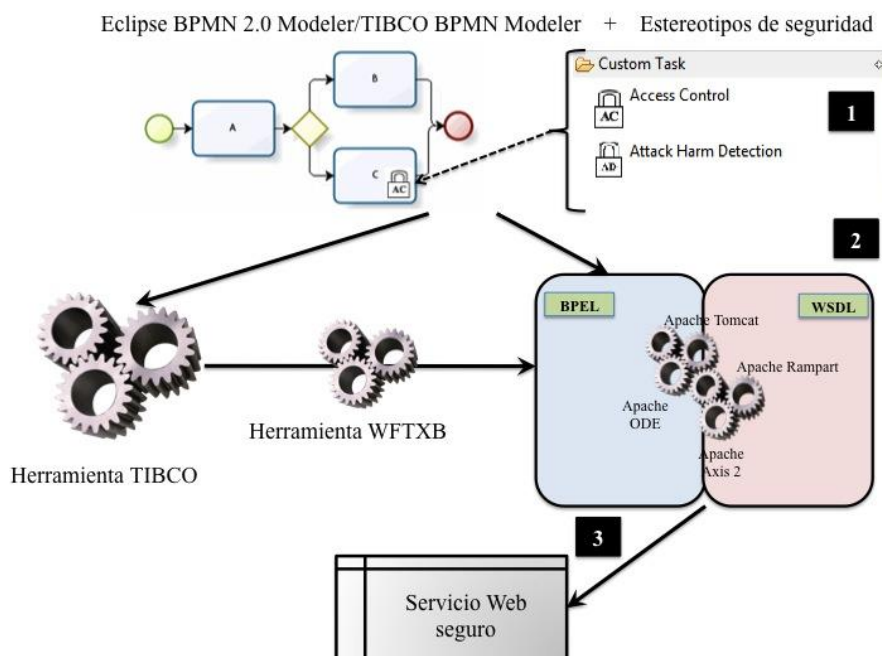


Figura 4.5: Recorrido para integrar la seguridad en BPEL desde un SBP

En la Figura 4.5 se pueden apreciar los recursos que se necesitan tanto para la integrar la seguridad en XPDL como en BPEL. En la sección 4.1 se determinó que para integrar la seguridad en XPDL se necesita de los atributos extendidos. La herramienta TIBCO [54] es la más adecuada para realizar tal procedimiento (en el siguiente capítulo, se hablará en mayor detalle sobre esta herramienta). Siguiendo la misma línea, en el capítulo dos, se mencionó la relación entre XPDL y BPEL, siendo la herramienta WFTXB la propuesta para transformar un código XPDL a BPEL. Del mismo modo, en esta subsección, se explicará la propuesta de integración de la seguridad en BPEL desde un SBP para obtener un Servicio Web seguro. Este camino consta de tres bloques: el bloque 1 contiene el diagrama BPMN seguro y los estereotipos de seguridad (los que se han denominado *SecurityStereotypes*). Éstos contienen la referencia de los requerimientos de seguridad definidos en un SBP que son interpretados en el diagrama BPMN, para esta tesis, esta adaptación contiene la referencia de Access Control, la cual se ve reflejada con el candado con las iniciales AC en su interior. Siguiendo el orden descendente, el bloque 2 contiene los lenguajes BPEL/WSDL y las correspondientes tecnologías que hacen posible la integración de la seguridad en BPEL. Por último, el bloque 3 contiene el Servicio Web seguro, el cual posee Access Control en sus propiedades. A continuación, se explicará cada bloque descrito en la Figura 4.5:

- **Bloque 1.** En primer lugar, se extendió el entorno de trabajo utilizado por Eclipse Ganymede [22] para crear diagramas BPMN llamado Eclipse BPMN Modeler [21] para que considerase una aplicación drag and drop (que fue desarrollado en el marco de esta tesis), la que permite incluir los requerimientos de seguridad en BPMN, permitiendo crear una descripción de un SBP en Eclipse BPMN Modeler. Una vez que se haya creado el diagrama BPMN seguro, se utiliza la propuesta de Ouyang et al. [63] (re-elaborada por García-Bañuelos [25]) llamada BPMN2BPEL para crear el correspondiente código BPEL desde el BPMN, en este caso, seguro. Cabe destacar que no fue necesario modificar BPMN2BPEL ya que lo que interesa es obtener el código BPEL en donde menciona el requerimiento de seguridad.
- **Bloque 2.** Una vez obtenido el código BPEL considerando los requerimientos de seguridad, se ejecuta el lenguaje BPEL bajo el entorno de ejecución Apache ODE [6] y se puede ver el modelo BPEL obtenido del código a través del editor BPEL para Eclipse Ganymede llamado Eclipse BPEL Editor [20]. A partir del código BPEL, se obtiene el archivo WSDL [83] (Web Service Description Language), donde están los atributos que serán utilizados para las especificaciones de seguridad y posteriormente se le agregan las políticas y estándares de seguridad de WS-Security y Policy [59]. Previo a lo anterior, se verificará si existen similitudes entre cada requerimiento de seguridad descrito en [68] con los estándares WS-Security, por lo que la inserción de la política o estándar de seguridad, dependerá de qué requerimiento se ingresó al diagrama BPMN. La tecnología que ejecuta los estándares de seguridad es Apache Rampart [71], el cual usa Apache Axis2 [51] para su funcionamiento.
- **Bloque 3.** Una vez obtenido el correspondiente código WSDL con las políticas y estándares de seguridad considerados, se creará un Cliente Web con Apache Tomcat 6 [81], el cual hará

una solicitud de un Servicio seguro.

En los puntos anteriores, se han descrito los tres bloques que posibilitan la integración de la seguridad hacia BPEL desde un SBP. A continuación, se mencionará una descripción en detalle por cada bloque.

4.2.2.1. Creación del BPMN Seguro (Bloque 1)

Como se puede observar en la Figura 4.5, el camino propuesto para la investigación es la ejecución sincronizada de varias tecnologías. Como primer paso, hay que considerar la notación BPMN del entorno Eclipse. Eclipse BPMN Modeler usa un modelo de objetos para lograr la apariencia de la notación BPMN en lugar de obligar a un esquema que describa completamente la especificación, con el objetivo de minimizar el impacto en las especificaciones desarrolladas.

Para la creación del BPMN seguro es necesario extender la funcionalidad del modelador BPMN de Eclipse. Para ello, se creó el plugin que ha sido denominado *SecurityStereotypes*, que almacena los requerimientos de seguridad y posibilita que el analista de negocio pueda hacer la acción de drag and drop sobre los requerimientos de seguridad que estime conveniente. De este modo, el modelador BPMN de Eclipse contendrá un plugin el cual dará la posibilidad de especificar la seguridad en BPMN cuando el modelador del negocio lo estime conveniente. Para esta tesis se ha limitado solo al desarrollo del requerimiento de seguridad de Access Control, no obstante, cada requerimiento de SBP puede ser desarrollado en forma similar.

4.2.2.2. Creación de BPEL/WSDL desde BPMN Seguro (Bloque 2)

Una vez creado el BPMN seguro usando Eclipse BPMN Modeler, se obtiene el correspondiente código y modelo BPEL usando para ello Eclipse BPEL Editor [20]. El objetivo de este programa es agregar soporte completo para Eclipse para la definición, creación, edición, implementación, prueba y depuración de procesos BPEL. Por lo anterior, el editor BPEL de Eclipse ofrece todas las facilidades tecnológicas para la creación y manipulación del código BPEL. Posteriormente, se usa la propuesta de Ouyang et al. [62] [63] [64] llamada BPMN2BPEL, que es una herramienta para la creación de archivos BPEL desde un diagrama BPMN. A pesar de existir entornos como Oracle [18], que posee una herramienta de transformación desde un BPMN a BPEL, se ha optado por BPMN2BPEL porque es de libre acceso. Ya que el diagrama contiene información adicional de la seguridad que después se ve reflejada en el código BPEL, no fue necesario modificar la herramienta BPMN2BPEL, los autores que han trabajado con la herramienta proponen lo anterior [77]. Una vez que se obtiene el código BPEL, hay que tener un entorno de ejecución para que funcione. Para el caso de BPEL, el entorno predilecto es Apache ODE [6] [77] [89]. Este motor de orquestación ejecuta los Procesos de Negocio siguiendo el estándar WS-BPEL, además, interactúa con los Servicios Web mediante el envío y recibo de mensajes, manipulación de datos y recuperación de errores.

El objetivo principal en el desarrollo de Apache ODE es la creación de un componente fiable, compacto y embedable capaz de gestionar la ejecución de los procesos empresariales de larga duración que se define utilizando el lenguaje BPEL. Los componentes claves de la arquitectura de ODE incluyen [43]:

- ODE BPEL Compiler
- ODE BPEL Runtime Engine
- ODE Data Access Objects (DAO)
- ODE Integration Layers (IL's)
- Herramientas de usuario.

Es importante explicar la razón por la que el lenguaje BPEL en sí, no se puede considerar la seguridad desde un Proceso de Negocio Seguro. El motor Apache ODE, donde se ejecuta BPEL, no soporta extensiones de seguridad. Lessen [43] menciona que la creación de Apache ODE, por ahora, no ha considerado extender el motor de ejecución a aspectos de seguridad. Las extensiones posibles para Apache ODE (versión 1.3.5, que es la última versión disponible) son [6]: Implicit Correlations, XPath Extensions, External Variables, entre otras. Cuando la versión 2.0 esté disponible, considerará la extensión de actividades que pueden soportar conexiones con estándares de seguridad para Servicios Web.

En este punto se ha explicado cómo se puede obtener el código BPEL desde un BPMN seguro. También se recalcó el por qué el lenguaje BPEL (por tema de versiones y actualizaciones) no soporta extensiones de seguridad. Es por eso que autores [77] [50] [90] que han trabajado con BPEL y Servicios Web, utilizan Web Services Description Language (WSDL) [83] para describir la seguridad.

La importancia de WSDL es que, junto con los elementos descritos anteriormente, hacen que BPEL pueda adoptar los Servicios Web como su mecanismo de comunicación externa (red). Así las facilidades de mensajería BPEL dependen del uso del WSDL para describir los mensajes entrantes y, por otro lado, permitir la especificaciones de Web Service Security (WS-Security). Estas especificaciones permitirán traducir los requerimientos de seguridad definidos en [68] en requerimientos de seguridad Web y obtener el correspondiente Servicio Web seguro. Los WS-Security son protocolos de comunicaciones que suministran un medio para aplicar seguridad a los Servicios Web, publicados por OASIS [58]. A su vez, estos estándares están contenidos en Apache Rampart [7], que es el módulo de seguridad de Apache Axis2 [5].

Apache Axis2 es un entorno de ejecución de Servicios Web, SOA y WSDL basado en Java, tanto para el cliente como el servidor de los Servicios Web. Apache Axis2 proporciona un modelo de objetos completo y una arquitectura modular que hace fácil agregar funcionalidad y soporte para los nuevos Servicios Web relacionados con las especificaciones y recomendaciones que el usuario desee.

Apache Axis2 permite realizar las siguientes tareas:

- Enviar mensajes SOAP
- Recibir y procesar mensajes SOAP
- Crear un Servicio Web de una clase Java
- Crear clases de implementación, tanto para el servidor v el cliente a través de WSDL
- Fácilmente recuperar el archivo WSDL para un servicio
- Enviar y recibir mensajes SOAP con archivos adjuntos
- Crear o utilizar un Servicio Web basado en REST
- Crear o utilizar los servicios con: WS-Security, WS-ReliableMessaging, WS-Addressing, WS-Coordinación v WS-Atomic.
- Utilizar la estructura modular de Axis2 para añadir fácilmente el apoyo de los nuevos requerimientos que surgen.

Apache Axis2 permite la manipulación de WS-Security, es por eso que se ha seleccionado esta herramienta para el desarrollo de la investigación. Como se mencionó, Axis2 dentro de sus elementos posee Apache Rampart, el cual se compone de un módulo básico y los paquetes relacionados con WS-Security y WS-Policy Security. Para el cifrado y firma XML, Apache Rampart utiliza OMXMLSecurity [4] y se utiliza OpenSSL [61] (Open Secure Sockets Layer) como la biblioteca de criptografía para OMXMLSecurity. Para obtener tanto Apache Rampart como Axis2 en Eclipse, se utilizó Eclipse Web Tools Platform para Eclipse Ganymede [23], ya que la herramienta BPMN2BPEL fue desarrollada para este ambiente.

Para continuar, es necesario asociar cada requerimiento de seguridad con alguna política o estándar de seguridad. Para ello, hay que recurrir a los estándares de seguridad existentes para Servicio Web. Cada requerimiento de seguridad definido en [68] cumple un rol y una funcionalidad específica, por otro lado, WS-Security está más orientado a la seguridad funcional del Servicio Web, por lo que la similitud que exista entre estos dos mecanismos de seguridad, no necesariamente tienen que ser compatibles, pero se pueden asemejar. Por lo tanto, en los siguientes puntos se hará una similitud entre los requerimientos de seguridad v los estándares WS-Security.

- **Non repudation:** para este requerimiento de seguridad se asociará el estándar de seguridad de Web Services Security: Non-Repudation [67]. Este requerimiento fue diseñado para cumplir cuatro requisitos:
 - La solicitud de mensajes SOAP debe ser capaz de solicitar una respuesta para el mensaje SOAP que se está transmitiendo.
 - La respuesta del mensaje SOAP debe ser capaz de enviar un respuesta de un mensaje SOAP que solicita un mensaje, ya sea incrustados en la respuesta de SOAP o en otro mensaje.

- Los solicitantes de mensajes SOAP deben ser capaces de especificar cuáles son los elementos del mensaje SOAP que desean tener firmados por el mensaje SOAP de respuesta.
 - Los mensajes recibidos SOAP deben ser capaces de transmitir una firma para los elementos que se pidió ser firmados por el solicitante de mensajes SOAP.
- **Attack Harm Detection:** para este requerimiento de seguridad, no existe especificación de WS-Security que cumpla con los requerimientos solicitados por Attack Harm Detection. Existe un trabajo realizado por Gruschka et al. [27] los cuales proponen una manera de proteger a los Servicios Web mediante mensajes de validación SOAP. Ellos proponen un firewall el cual contiene mensajes de validación SOAP, esquemas XML y tecnologías Java que hacen posible aplicar la protección contra ataques a los Servicios Web.
 - **Integrity:** para este mecanismo de seguridad, se ha optado por WS-Security Policy Integrity Assertion [55]. Para ello, existen dos mecanismos que están definidos para especificar el conjunto de las partes del mensaje a la integridad de la protección. Uno de ellos utiliza QNames para especificar encabezados de los mensajes o el cuerpo del mensaje, mientras que el otro utiliza expresiones XPath para identificar cualquier parte del mensaje.
 - **Privacy:** este requerimiento se asocia con Cross-Enterprise Security and Privacy Authorization (XSPA) [57], que es un perfil donde se especifica un conjunto de elementos que hacen posible asegurar la privacidad en los Servicios Web.
 - **Access Control:** este requerimiento de seguridad tiene su similitud con Web Service Security Username Token [56]. Este requerimiento de seguridad mediante un UsernameToken y un password es capaz de realizar un control de acceso eficiente para los Servicios Web.
 - **Security Roles y Permissions:** para estos requerimientos no existen especificaciones de WS-Security. La literatura recomienda realizar una política de seguridad usando WS-Policy [55] por cada caso en particular, ya que los negocios de las empresas siempre son distintos, por lo que no se pueden crear roles ni permisos de seguridad en general. La literatura propone crear políticas de seguridad basadas en RBAC (Role Based Access Control) y WS-Trust, ya que estos mecanismos de seguridad pueden ayudar a definir una correcta política de permisos y roles en los Servicios Web.

En la Tabla 4.1 se resumen los puntos anteriores.






Nombre	Notación	Asociación BPMN-BPD	Asociación WS-Security/Policy
Non Repudation		Flujos de mensaje	WS-Security: Non-Repudation
Attack Harm Detection		Pool, Lane, Grupo, Actividad, Objeto de datos, Flujo de secuencia	Propuesta de Gruschka et al. [27]
Integrity		Fujos de mensaje, objeto de datos	WS-Security Policy Integrity Assertion
Privacy		Pool, Lane, Grupo	Cross-Enterprise Security and Privacy Authorization
Access Control		Pool, Lane, Grupo, Actividad	WS-Security UserName Token
Security Roles/Permissions		Pool, lane, Grupo/ Actividad, Flujo de mensaje, Objeto de datos	WS-Security Policy

Tabla 4.1: Requerimientos de seguridad v su relación con BPMN-BPD v WS-Security/Policy

Se han establecido las semejanzas que tienen los requerimientos de seguridad con los estándares de WS-Security y Policy. Cabe destacar que cada estándar, tiene sus estrategias particulares que satisfacen el requerimiento. Por ello, una vez que se desarrolle el archivo WSDL con sus correspondientes estándares o políticas de seguridad, hay que ejecutar el archivo WSDL en un motor de ejecución para llevarlo a un Servicio Web seguro.

4.2.2.3. Creación del Servicio Web seguro (Bloque 3)

En este bloque, resta generar el Cliente Web para el Servicio Web seguro a partir del archivo WSDL. El entorno Eclipse Ganymede, en el cual se ha basado toda esta propuesta, ofrece alternativas tecnológicas (Eclipse Web Tools Platform) para generar el Cliente Web de manera automática. Una vez realizado lo anterior, es posible obtener un Servicio Web seguro con el requerimiento de seguridad Access Control a partir de un diagrama BPMN seguro. Este Servicio Web seguro contendrá información que limitará el acceso de usuario a partes del proceso, dependiendo del analista de negocio.

4.3. Resumen

En este capítulo se detalló el camino para la integración de la seguridad en XPDL/BPEL desde un Proceso de Negocio Seguro. El lenguaje XPDL es un lenguaje orientado a grafos con el objetivo de que la información que está en un diagrama BPMN pueda ser traspasado en

distintos entornos de desarrollo sin perder información. Para ello, en este capítulo se ha detallado lo siguiente:

- Para ingresar los requerimientos de seguridad en XPDL, se deben usar los atributos extendidos. Los atributos extendidos son elementos que pertenecen al lenguaje XPDL cuyo objetivo es enriquecer aún más la información que será almacenada y transmitida desde un entorno de desarrollo u otro.
- XPDL posee la capacidad de contener las referencias en dónde se utilizan los requerimientos de seguridad, pero no es capaz de procesar dichos requerimientos.

Lo anterior permite no sólo ingresar Access Control a XPDL, sino que todos los requerimientos de seguridad definidos en un SBP, ya que como se dijo en el párrafo anterior, XPDL traspasa el diagrama BPMN entre herramientas, pero no le interesa el cómo se ejecuta el diagrama. De eso se preocupa BPEL.

Por el lado de BPEL, se propone un camino usando tecnologías con el objetivo de que pueda ser expandible para agregar más requerimientos. La propuesta de esta investigación se resume en la Figura 4.4, donde se expone el esquema para llevar un BPMN seguro a un Servicio Web, posteriormente, en la Figura 4.5 se ilustra el recorrido de la propuesta, el cual está dividido en tres bloques. El bloque 1 se compone del diagrama BPMN seguro con los SecurityStereotypes, los cuales permiten al analista de negocios determinar dónde quiere colocar los candados de seguridad descrito en [68] para considerar la seguridad. Luego, en el bloque 2 se obtiene el código BPEL correspondiente del diagrama BPMN seguro, para posteriormente agregar las variables que consideran la seguridad. BPEL funciona bajo el motor Apache ODE el cual orquesta los servicios obtenidos del diagrama BPEL.

Teniendo el código BPEL con las variables que considerarán la seguridad, se desarrolla un archivo del tipo WSDL. En el archivo WSDL se deben agregar las políticas de seguridad que serán ocupadas por las variables que fueron definidas en BPEL. Una vez que las políticas de seguridad fueron agregadas, se deben asignar al código WSDL dependiendo del requerimiento de seguridad que se haya elegido. Para ello, en la sub-sección 4.2.2.2 se hizo una comparación entre los requerimientos de seguridad y su similar con respecto a WS-Security y Policy. Esta comparación, está basada en las propiedades que posee cada requerimiento de seguridad y las propiedades que ofrecen los estándares de seguridad, pero algunos no tienen traducción directa a un estándar de seguridad, por lo que se asocia a una política de seguridad, definida por la organización quien usa el proceso. Para utilizar las políticas de seguridad, el entorno de desarrollo debe poseer Apache Rampart, sub-módulo de Apache Axis2. Apache Rampart es el módulo open source que contiene todos los estándares y políticas de seguridad que se usan en esta propuesta. Una vez que todo esté listo, se utiliza Apache Tomcat para hacer funcionar el Servicio Web a partir del archivo WSDL. Junto con Apache Tomcat, se debe utilizar Apache ODE, quien es el orquestador de los Servicios Web. Como producto final, se obtiene un Servicio Web seguro que contiene la especificación de seguridad de Access Control

Capítulo 5

Ejemplo ilustrativo

En este capítulo se ilustrará la implementación de un ejemplo que describe un Proceso de Negocio seguro relacionado con la entrega de una ficha a un paciente que es atendido en un centro de atención de salud. Para realizar este diagrama, se ha observado el proceso que realizan los hospitales en general al momento de entregar una ficha y se generalizó para ser descrito en un SBP. Cabe mencionar que se han omitido Pool de gran importancia como por ejemplo Paciente, Empresa de Seguro, entre otros, ya que la herramienta BPMN2BPEL sólo transforma a código un proceso descrito en un solo Pool.

El proceso comienza cuando al servicio de admisión del centro de atención de salud le llega la solicitud de admisión del paciente para ser revisada. Posteriormente, se hace una recolección de datos que consiste en la verificación de la ficha clínica y la captura de información relacionada con los seguros. Una vez realizadas todas las operaciones de recolección de datos, se hacen los exámenes de evaluación al paciente y se rellena la ficha clínica. Luego, se completa la ficha con información del paciente, para ello, se debe ingresar a los datos sociales. El analista del negocio que crea el SBP ha determinado que la información social del paciente es privada, lo que involucra crear un control de acceso para acceder a esos datos. Como ya se tiene la información médica del paciente, se debe verificar si el paciente posee alguna deuda con el centro de atención de salud, en caso de que el paciente posea deuda, la debe cancelar y posteriormente se verifica si el paciente trabaja o no. En caso de que no trabaje, se debe ingresar que éste está en un riesgo social y se entrega la ficha clínica con esa información. Si el paciente trabaja, se le entrega la ficha clínica sin ninguna observación de riesgo social. En caso de que el paciente no posea ninguna deuda, se le entrega la ficha clínica. El proceso de verificación de riesgo social desde el punto de vista de la deuda, se hace en el departamento de contabilidad y para manipular los datos de los pacientes, se debe autenticar en el sistema. En el SBP, el control de acceso para permitir la información social del paciente se ve reflejado en la actividad Rellenar información del paciente, la cual está con un candado que describe la notación definida en un SBP para señalar que tal actividad nosee un control de acceso. El proceso descrito anteriormente se muestra en la Figura 5.1.

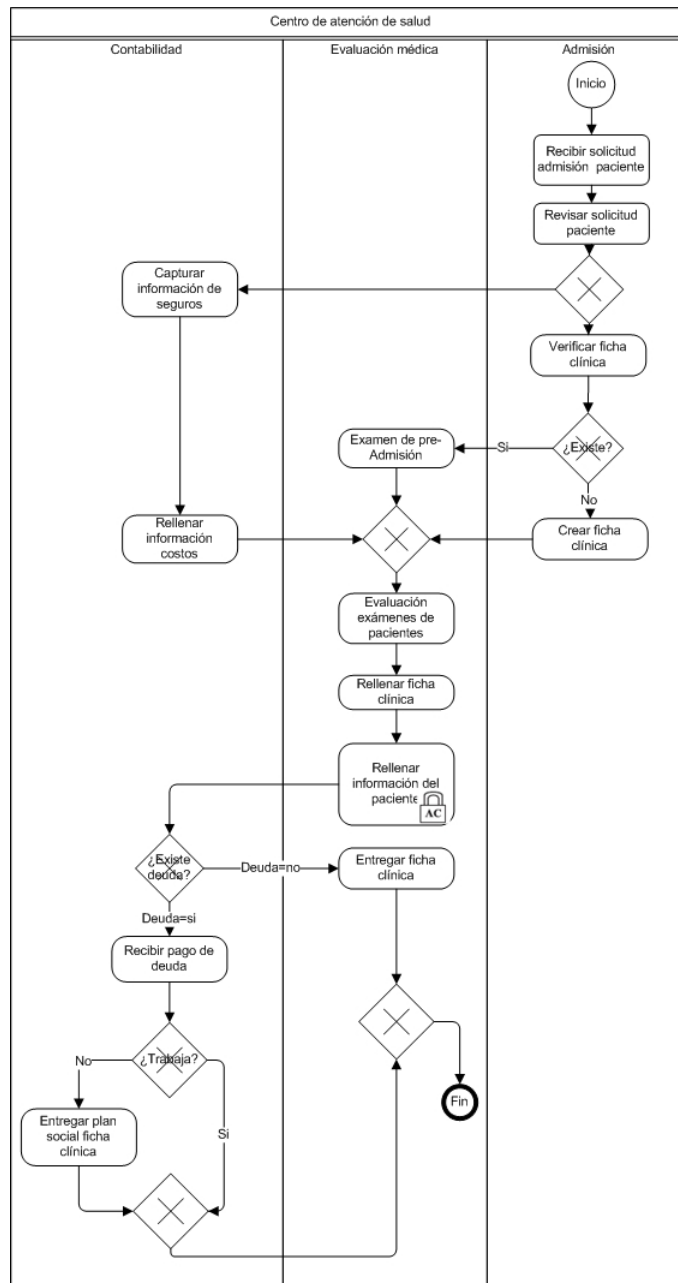


Figura 5.1: SBP de entrega de ficha a paciente en un centro de atención de salud

El objetivo de desarrollar este ejemplo es para comprobar que desde un SBP es posible obtener el código XPD L que describa la seguridad y un Servicio Web seguro utilizando el lenguaje BPEL. Para ello, a partir de la Figura 5.1, se desarrollará el Control de Acceso para la actividad Rellenar información del paciente.

5.1. Implementación de la integración de la seguridad en XPDL

En esta sección, se mostrará parte del código XPDL que se obtiene del SBP que se ha mostrado en la Figura 5.1 en que se ha especificado Access Control en la actividad Rellenar información del paciente (el código completo de la Figura 5.1 se aprecia en el Anexo A). Dentro de las ofertas de herramientas disponibles para el desarrollo de diagramas BPMN con su correspondiente código XPDL, se ha seleccionado TIBCO versión 3.2 [54], ya que es una herramienta basada en el entorno de desarrollo Eclipse Ganymede [22], lo que posibilita el acceso al código fuente para poder ingresar extensiones. La Figura 5.2 corresponde a la actividad Rellenar información del paciente.

```
1:      <xpdl2:Activity Id="..."
2:      Name="Rellenar información del paciente" xpdExt:Visibility="Private"
3:      xpdExt:DisplayName="Rellenar información del paciente">
4:          <xpdl2:Description>Esta actividad requiere un control de acceso
5:      </xpdl2:Description>
6:      <xpdl2:Implementation>
7:          <xpdl2:No/>
8:      </xpdl2:Implementation>
9:      <xpdl2:ExtendedAttributes>
10:         <xpdl2:ExtendedAttribute Name="Access Control"
11:         Value="1"></xpdl2:ExtendedAttribute>
12:     </xpdl2:ExtendedAttributes>
13:     ...
14: </xpdl2:Activity>
```

Figura 5.2: Código XPDL de rellenar la ficha clínica con el requerimiento de seguridad Access Control

En la actividad de Rellenar información del paciente del Pool Evaluación médica, se ha agregado el requerimiento de seguridad de Access Control en la línea 4 y 10, donde se detalla una pequeña porción de la descripción del requerimiento. De esta manera, se muestra cómo se pueden agregar requerimientos de seguridad en XPDL. Del mismo modo, se puede comprender el por qué de la capacidad limitada de XPDL para tratar con requerimientos de seguridad. Los requerimientos de seguridad descritos en [68] son mecanismos para crear un BPMN seguro y tales requisitos utilizan varias técnicas y procedimientos que XPDL no es capaz de procesar. Para el ejemplo que se está analizando, el archivo .xpdl en donde se detalla el proceso de préstamos contendrá la información de seguridad, manteniendo este requerimiento cuando el código se traspase de un entorno de desarrollo en otro.

Como va se diiera anteriormente. la herramienta que se usará para trabajar con XPDL es

TIBCO, ya que es de libre acceso y permite la conexión con entornos de Eclipse y sus respectivas aplicaciones. A pesar de que hay herramientas tales como Together XPDL [80], éstas no poseen la flexibilidad para agregar otros componentes al entorno de desarrollo. Adicionalmente, TIBCO posee una interfaz de usuario al momento de especificar atributos extendidos. En la Figura 5.3, se puede observar que el analista de negocios puede detallar una descripción del mecanismo de seguridad que se encuentra en una cierta actividad. En el caso del mecanismo de seguridad se encuentre en la Web y que pueda ser interpretado por XPDL, el analista de negocios puede detallar la URL de la página y XPDL es capaz de referenciarla en su código.

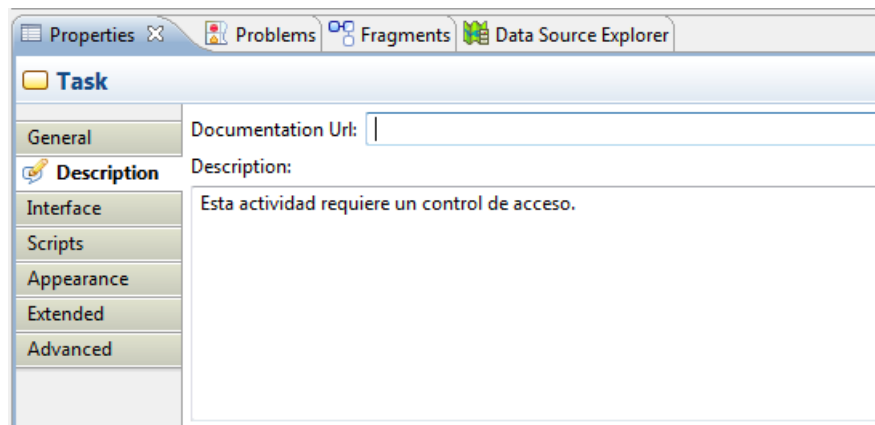


Figura 5.3: Interfaz de TIBCO para describir en detalle las actividades

Por otro lado, al momento de especificar los atributos extendidos para agregar la seguridad, TIBCO también da la posibilidad de detallar el mecanismo de seguridad deseado como se puede ver la Figura 5.4.

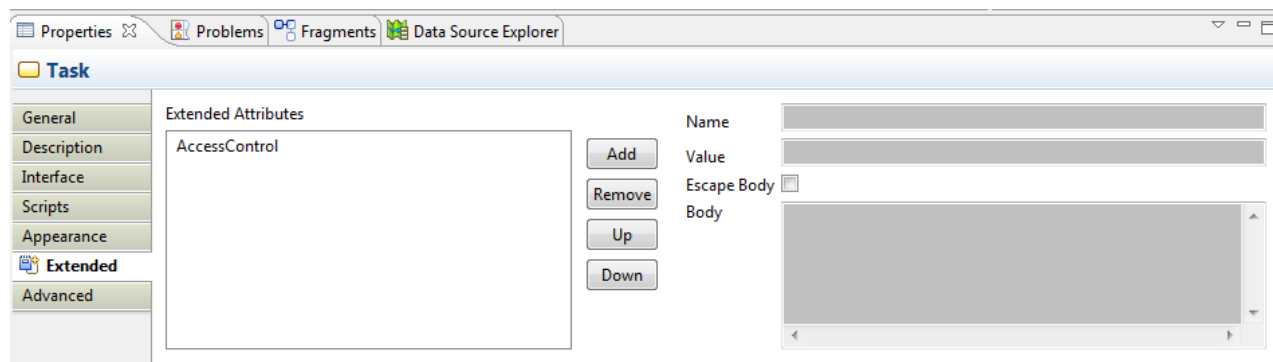


Figura 5.4: Interfaz de TIBCO para describir los atributos extendidos

Por ahora, TIBCO no ofrece la posibilidad de extender el entorno de desarrollo que posee para describir diagramas de BPMN. A pesar de que TIBCO está basado en Eclipse Ganymede, no ocupa Eclipse BPMN Modeler para generar sus diagramas BPMN, utiliza su propio modelador.

lo que limita su extensión para agregar los componentes visibles de requerimientos de seguridad (candados).

5.2. Implementación para integrar la seguridad en BPEL

En esta sección se utilizará como esquema la Figura 4.5 de la sección 4.2 del capítulo 4, la cual muestra el recorrido que se debe hacer para obtener el Servicio Web seguro a partir del SBP. A continuación en la Figura 5.5 se presentará el esquema de la Figura 4.5 junto con los códigos más importantes que se utilizarán para la implementación de este ejemplo ilustrativo.

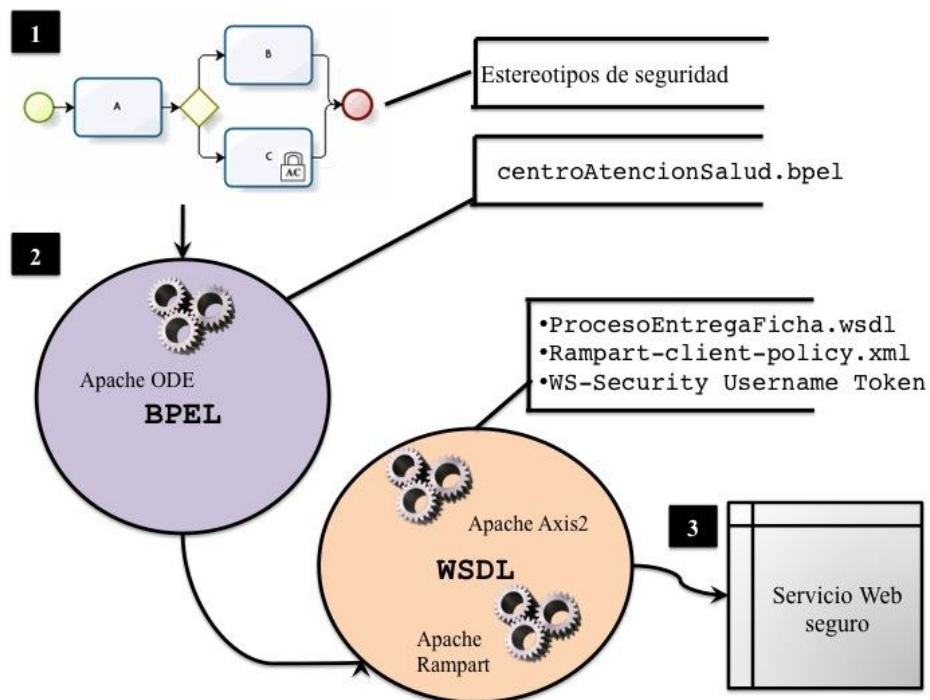


Figura 5.5: Recorrido de transformación a un Web Service Seguro desde un SBP mostrando los principales elementos utilizados

5.2.1. Creación de los estereotipos de seguridad (Bloque 1)

Para la creación de los estereotipos de seguridad, hay que analizar las posibilidades de expansión del modelador de BPMN. Éste posee un metamodelo del cual se derivan todas las clases que hacen funcionar al programa. Se intentó expandir este metamodelo con el metamodelo de requerimientos de seguridad, pero la expansión de cada requerimiento de seguridad involucra un desarrollo que consume gran cantidad de tiempo. Por otro lado, la inestabilidad del modelador BPMN de Eclipse Ganymede, no permite extender propiedades para especificar los requerimientos de seguridad. Es por eso que se optó por la creación de un plugin que tuviera la acción de drag

and drop para insertar los requerimientos de seguridad en el modelador BPMN 2.0 de Eclipse Indigo, realizado en el Sistema Operativo Windows 7 [19].

Luego de crear el plugin, se programaron las clases Java necesarias para la creación e inserción de los requerimientos de seguridad, los cuales quedan almacenados en la carpeta Custom Task en la Paleta BPMN 2.0. Una vez terminada esta tarea, Eclipse Indigo cuenta con un plugin que contiene los estereotipos de seguridad (ver Figura 5.5, bloque 1), como se ve en la Figura 5.6. De esta manera, se obtiene la Figura 5.6. Para esta tesis, se ha implementado el requerimiento de seguridad de Access Control y, a manera de expandir más el plugin, el requerimiento de Attack Harm Detection.

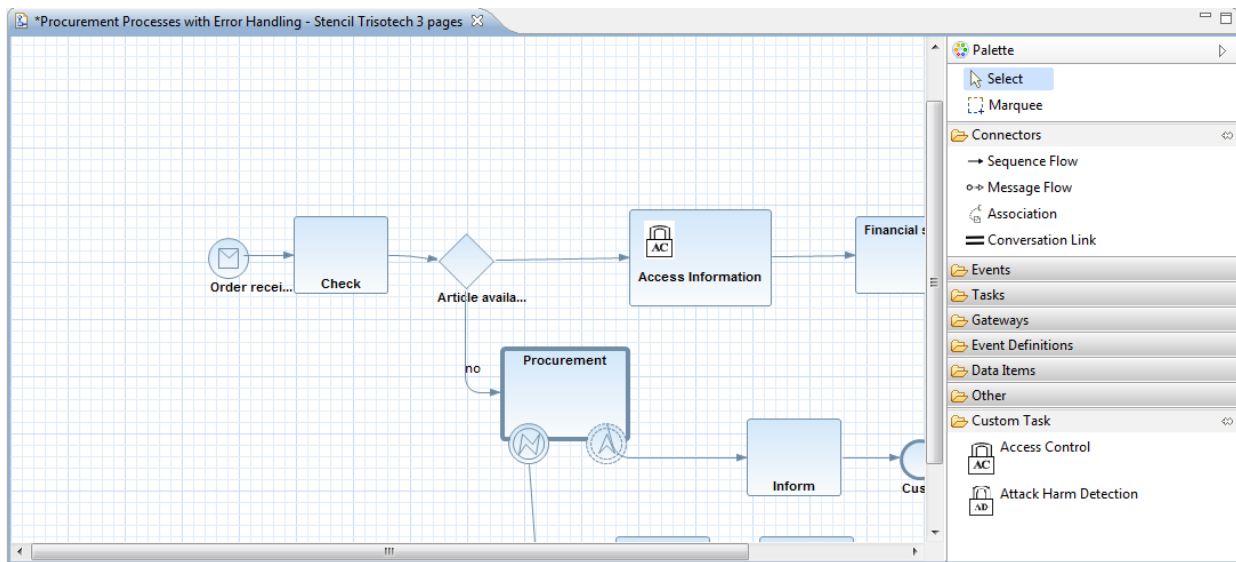


Figura 5.6: Estereotipos de seguridad en Eclipse Indigo

5.2.2. Obtención del código BPEL y archivos WSDL (Bloque 2)

Una vez que el diagrama BPMN seguro haya sido creado, se utiliza la herramienta BPMN2BPEL para generar el correspondiente código y modelo BPEL. Para esta tesis, se ha desarrollado una parte de la Figura 5.1, ya que cada actividad en el diagrama BPMN seguro es un Servicio Web independiente, por lo tanto, este capítulo se enfocará en el desarrollo de la actividad del diagrama BPMN seguro que posee el requerimiento de seguridad de Access Control (ver Figura 5.1) la cual es Rellenar información del paciente. En la Figura 5.7 se muestra el modelo BPEL de la Figura 5.1. El diagrama BPMN seguro se ha transformado en un modelo orientado a bloques llamado centroAtencionSalud.bpel (ver Figura 5.5, bloque 2). El modelo original que se ha obtenido, se ha mejorado agregando detalles para su correcto funcionamiento, ya que la herramienta BPMN2BPEL sólo genera el esquema BPEL del diagrama BPMN seguro. Al igual que un diagrama BPMN, este modelo tiene un inicio y un fin, donde cada gateway del BPMN seguro es una condición **If** de BPEL y cada actividad del diagrama tal como Rellenar información del paciente, es un Assign en BPEL. El código BPEL centroAtencionSalud.bpel contiene todos

los elementos para que funcione el motor de ejecución Apache ODE (el código completo de la Figura 5.7 se aprecia en el Anexo B).

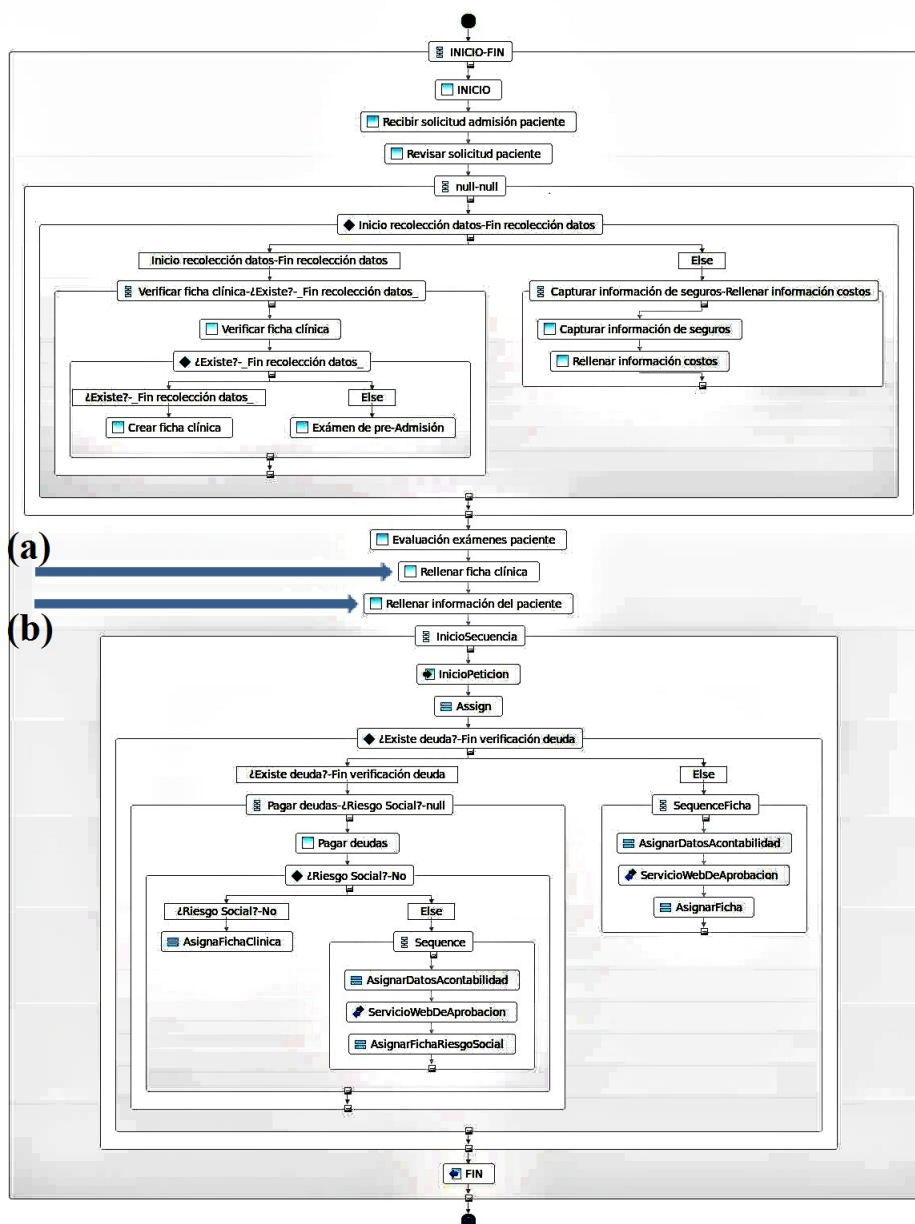


Figura 5.7: Modelo BPEL obtenido del BPMN seguro de Proceso de entrega ficha usando la herramienta BPMN2BPEL

En la Figura 5.7, se observan dos flechas (a) y (b). La flecha (a) describe una actividad del SBP que no ha sido implementada, la cual se refleja en la Figura 5.8

```

...
1:<bpel:empty name="Rellenar ficha clínica"/>
...

```

Figura 5.8: Trozo de código BPEL de la actividad Rellenar ficha clínica donde no se especifica seguridad

En la Figura 5.8 se ilustra parte del código BPEL de la actividad Rellenar ficha clínica generado por BPMN2BPEL. Como esta actividad no ha sido implementada, BPMN2BPEL genera la actividad como vacía (empty), por lo que no posee ninguna funcionalidad. En cambio, en la flecha (b) se muestra la actividad Rellenar información del paciente, cuyo encabezado de código se ve reflejado en la Figura 5.9.

```

...
1:<bpel:sequence name="Rellenar información del paciente">
2:  <bpel:receive name="InicioPeticion"... variable="entrada" ...</bpel:receive>
3:  <bpel:condition><![CDATA[$entrada.AccessControl]]></bpel:condition>
...

```

Figura 5.9: Trozo de código BPEL de la actividad Rellenar ficha clínica

En la Figura 5.9, se puede apreciar la diferencia cuando una actividad posee el requerimiento de Access Control. El código BPEL generado por la herramienta BPMN2BPEL permite especificar mediante una variable (que en esta implementación se llama entrada) que la actividad posee el requerimiento de seguridad de Access Control. Esta variable está declarada en la línea 3 de la Figura 5.9. Es así que, cuando una actividad posee un requerimiento de seguridad, BPMN2BPEL generará el código BPEL detallando mediante una variable que describa el requerimiento anteriormente señalado. Por ahora, la variable entrada se ha ingresado de forma manual, ya que como se ha dicho, el modelador BPMN de Eclipse Ganymede no permite extender el modelador a otras propiedades, razón por la cuál se optado por trabajar con el modelador BPMN 2.0 de Eclipse Indigo.

Posteriormente, en la Figura 5.10 se muestra los partnerlink (líneas 6 y 9 destacadas) que se han creado en este ejemplo, estos son paciente y entregaFicha. Ellos serán los elementos que interactuarán con el código BPEL. En otras palabras, el paciente pedirá su ficha clínica y entregaFicha mostrará un mensaje si la ficha se entrega con datos de riesgo social o no.

```

1:<bpel:process exitOnStandardFault="yes"
2:name="centroAtencionSalud"
3:...
4:importType="http://schemas.xmlsoap.org/wsdl/"></bpel:import>
5:<bpel:partnerLinks>
6:  <bpel:partnerLink name="paciente"
7:    partnerLinkType="tns:ProcesoDeEntregaPLT"
8:    myRole="ProcesoDeEntregaProvider"></bpel:partnerLink>
9:  <bpel:partnerLink name="entregaFicha"
10:...
11:</bpel:partnerLink>
12:</bpel:partnerLinks>

```

Figura 5.10: Trozo de código BPEL del proceso de préstamo

Luego, se declaran las variables que interactuarán con el código BPEL. Para el ejemplo, se crearán las siguientes variables: PrimerNombre, PrimerApellido, Empleado_si_no, Deuda, Login y Password (ver desde la línea 5 hasta la 10 de la Figura 5.11).

```

1:<bpel:literal
2:xml:space="preserve"><tns:ProcesoDePrestamoRequest
3:xmlns:tns="http://cnti_calidad/procesoDePrestamo"
4:xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
5:  <tns:PrimerNombre></tns:PrimerNombre>
6:  <tns:PrimerApellido></tns:PrimerApellido>
7:  <tns:Empleado_si_no></tns:Empleado_si_no>
8:  <tns:Deuda></tns:Deuda>
9:  <tns:Login></tns:Login>
10:  <tns>Password></tns>Password>
11:</tns:ProcesoDePrestamoRequest> </bpel:literal>

```

Figura 5.11: Trozo de código BPEL donde se detallan las variables que se utilizarán en el Servicio Web

Una vez que se crean todas las variables necesarias y, por otro lado, configurados los elementos necesarios para BPEL, se puede pasar a la siguiente etapa en que se trabajará con los archivos WSDL. Una vez creado el archivo BPEL, hay que obtener el archivo WSDL, que para este ejemplo, ya se ha desarrollado previamente. Posteriormente, hay que agregar la política de seguridad al

archivo, de manera que cuando se desarrolle el Servicio Web seguro, pueda hacer uso de éstas. Con los pasos anteriores, se ha obtenido un archivo llamado `ProcesoDeEntregaFicha.wsdl` (ver Figura 5.5, bloque 2), donde se encuentran especificados los estándares de seguridad de UsernameToken. Una vez definida la política de seguridad, se declara la utilización de UsernameToken para ser usado dentro del código WSDL (ver Figura 5.12. líneas 6. 9 v 10).

```

1: <wsp:Policy wsu:Id="UsernameToken" xmlns:wsu=
2:   "http://docs.oasis-open.org/wss/2004/01/
3:   oasis-200401-wss-wssecurity-utility-1.0.xsd"
4:   ...
5:   <sp:SupportingTokens
6:     xmlns:sp="http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702">
7:       <wsp:Policy>
8:         <sp:UsernameToken sp:IncludeToken=
9:           "http://docs.oasis-open.org/ws-sx/ws-securitypolicy
10:          /200702/IncludeToken/AlwaysToRecipient"/>
11:       ...
12:     </wsp:Policy>

```

Figura 5.12: Política de seguridad de Access Control

En la línea 8 se declaran los esquemas de UsernameToken. Luego, entre las líneas 10 y 11 se declaran los tokens que serán utilizados por UsernameToken. Después que se ha declarado tanto la política de seguridad como el mecanismo de UsernameToken, es necesario especificar en el código WSDL dónde se quieren utilizar estas políticas de seguridad (ver Figura 5.13).

```

1:<operation name="peticion">
2:...
3: <wsp:PolicyReference xmlns:wsp=
4:   "http://schemas.xmlsoap.org/ws/2004/09/policy" URI="#UsernameToken"/>
5:...
6: <soap:operation
7:   <wsp:PolicyReference xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
8:     URI="#SignOnly"/>
9:   <soap:body use="literal" />
10: </input>

```

Figura 5.13: Código WSDL donde se detalla el ingreso de la seguridad

En el ejemplo, estas políticas se declaran en operación de petición (ver Figura 5.13, líneas 4 y 7) y en las variables de entrada de la misma operación. Una vez hecho lo anterior, es posible crear el Servicio Web seguro. Cabe destacar que las variables Login y Password se han creado como mecanismos de control de acceso, con el objetivo de usar el UsernameToken para la autenticación. En el siguiente bloque, se desarrollará Servicio Web de prueba para comprobar los pasos anteriores.

5.2.3. Creación del Servicio Web seguro (Bloque 3)

En este bloque, se realizan los últimos pasos para crear el Servicio Web seguro. El entorno Eclipse Web Tool Platforms ofrece una alternativa de creación de un Cliente Web a partir de un archivo WSDL, que en este ejemplo es `ProcesoDeEntregaFicha.wsdl`. Esta alternativa es usada por los desarrolladores para probar sus Servicios Web de manera automática, cuando no es necesaria ninguna modificación en el código WSDL. En el caso del ejemplo de esta tesis, sí hay que modificar el código agregando propiedades, las cuales son los WS-Security y Policy. Por lo tanto, hay que realizar lo siguiente:

- Desarrollar clases Java para la creación y prueba del Servicio Web seguro, para lo cual se desarrolló la clase Java `PruebaWebServiceSeguro.java`.
- Hay que crear otra clase llamada `PolicyTool.java`, la cual se encarga de administrar las llamadas a los estándares de seguridad cuando se necesite.
- Además de las clases Java mencionadas, hay que agregar los archivos XML de las políticas de seguridad. Para esta tesis se ha desarrollado `rampart-client-policy.xml` (ver Figura 5.5, bloque 2), archivo que debe estar dentro de la carpeta en donde se desarrolle el proyecto. Este archivo XML contendrá la información de user y password (la variable password se encuentra en la clase java `PWCHandler`) que se utilizará en la llamada del Servicio Web seguro (ver Figura 5.14, líneas 9 v 10).

```

1:<wsp:Policy xmlns:wsu=
2:  "http://docs.oasis-open.org/wss/2004/01/
3:oasis-200401-wss-wssecurity-utility-1.0.xsd"
4:  xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy">
5:  <wsp:ExactlyOne>
6:    <wsp:All>
7:      <ramp:RampartConfig xmlns:ramp=
8:        "http://ws.apache.org/rampart/policy">
9:        <ramp:user>alice0112</ramp:user>
10:    <ramp:passwordCallbackClass>PWCHandler</ramp:passwordCallbackClass>
11:...
12:</wsp:Policy>

```

Figura 5.14: Código de rampart-client-policy.xml

Una vez que se hayan definido y configurado todos los elementos para hacer funcionar el Servicio Web seguro, se necesita del funcionamiento de dos servidores al mismo tiempo: Apache Tomcat 6.0 y Apache ODE, ya que el primero ejecuta los Servicios Web en navegador Web y el segundo coordina la ejecución de los Servicios Web. En estos servidores es posible crear un Servicio Web seguro (ver Figura 5.5, bloque 3) a partir de la clase PruebaWebServiceSeguro.java. En el siguiente trozo de código se ilustra la conexión con Apache Rampart y se establece que el login sea "alice0112" y el password sea "fsmo28011993" (Ver Figura 5.15, líneas 6 y 7).

```

1: ServiceClient client = servicio._getServiceClient();
2:   client.getAxisService().applyPolicy(loadPolicy
3:   ("rampart-client-policy.xml"));
4:   client.engageModule("rampart");
5:
6:   petition.setLogin("alice0112");
7:   petition.setPassword("fsmo28011993");

```

Figura 5.15: Fragmento de clase Java para la administración de la creación del Servicio Web seguro

La ultima tarea es solicitar el Servicio Web seguro. En la Figura 5.16 se muestra la pantalla inicial del Servicio Web.

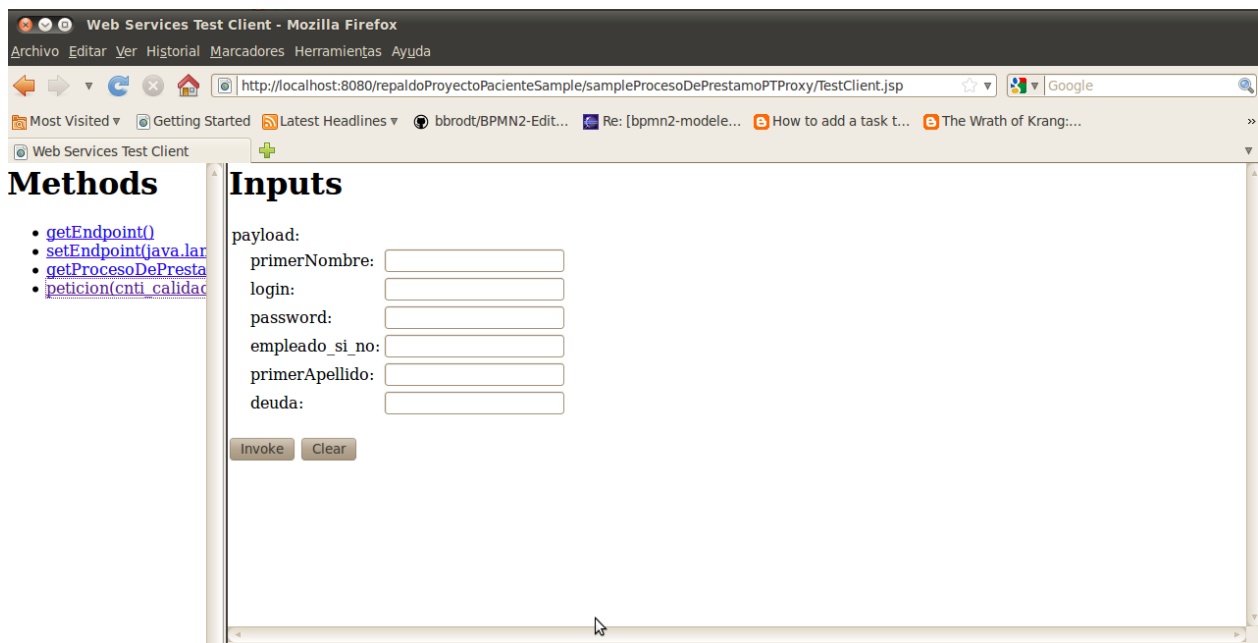


Figura 5.16: Inicio de Servicio Web seguro entrega ficha

En la Figura 5.16 se observan las variables: PrimerNombre, Login, Password, Empleado_si_no, PrimerApellido y Deuda. A estas variables se le asignará valores para probar el Servicio Web seguro. El Login y Password están condicionados en la política de seguridad que se describió en los párrafos anteriores. Cuando se ingresan todos los datos correspondientes a las variables descritas, se obtiene un Servicio Web donde se detalla un mensaje que dice si una ficha es entregada de forma normal o con riesgo social.

Resumiendo, se ha creado un diagrama BPMN al cual se le agregan los estereotipos de seguridad. Luego, a partir del diagrama BPMN seguro, se obtiene el correspondiente código BPEL que contiene la información de seguridad desde el diagrama BPMN seguro. A continuación, se crean las variables que serán probadas en el Servicio Web y se crea el archivo WSDL a partir del código BPEL. Una vez obtenido el código WSDL, se le integra los estándares WS-Security y Policy y al final, se genera el Servicio Web a partir del código WSDL.

5.3. Principales dificultades

En esta sección se explicarán las principales dificultades que se presentaron en el desarrollo de la implementación de la investigación. Autores [24] [77] que han trabajado con las herramientas descritas así lo respaldan.

- **Tecnologías:** Una de las grandes dificultades fueron las tecnologías utilizadas. La base de esta propuesta de investigación fue el plugin BPMN2BPEL. Este plugin fue probado en todas las versiones de Eclipse, desde Eclipse Europa hasta Indigo, pero la única versión en la cual

funcionó la herramienta fue Eclipse Ganymede, cuya última versión fue lanzada en el año 2009.

Esto condicionó la solución propuesta al uso de tecnologías que fueran compatibles con esta versión de Eclipse. Todos los plugin y programas que fueron ocupados para esta tesis, en un principio, estaban orientados a las últimas versiones de Eclipse, por ejemplo, si se utilizaba la última versión de Eclipse BPEL Editor, BPMN2BPEL no funcionaba, o si se usaba la notación BPMN 2.0 para Eclipse, no funcionaba BPMN2BPEL y tampoco Eclipse Ganymede, entre otras cosas más.

Por otro lado, los servidores también se convirtieron en dificultades para esta propuesta. En un principio, el funcionamiento de los servidores por separado no habían mostrado ningún problema, pero para hacer funcionar la propuesta descrita en la Figura 5.5, se tiene que tener en funcionamiento ambos servidores, Apache Tomcat y ODE.

Otra dificultad, extender programas. Para crear un BPMN seguro, se creó el plugin SecurityStereotypes el cual tenía la función de almacenar los estereotipos de seguridad. El problema de la creación de este plugin básicamente radica en la extensión de Eclipse BPMN Modeler versión 1.0, ya que cualquier elemento que se ingrese al modelador, éste deja de funcionar, por problemas de versión. Al consultar con los creadores de Eclipse BPMN Modeler, tampoco se pudo encontrar una solución al problema. Es por eso que se optó por Eclipse Indigo, ya que posee el modelador BPMN 2.0 que permitió la creación de un plugin que posee los requerimientos de seguridad.

- **Estándares de seguridad:** Otro problema que se presentó en esta investigación, fue el acceso a los estándares de seguridad de WS-Security y Policy. A pesar de que son estándares open source, fue difícil encontrar el acceso a ellos y, mucho más, a su especificación. Por otro lado, el acceso a Apache Rampart (módulo de seguridad) también fue difícil debido a que ambos, Apache Rampart y los estándares de seguridad, se encontraban en repositorios virtuales. El principal problema de los repositorios virtuales son dos:

- Dónde descargar la información de los repositorios.
- La estabilidad del funcionamiento de ellos.

El primer punto se refiere que si un entorno de desarrollo quiere tener acceso a estos archivos que se encuentran en los repositorios, debe tener los programas ad hoc para descargar la información. Afortunadamente, Eclipse Ganymede sí tenía la tecnología compatible para descargar la información. El segundo punto se refiere a que hubo veces en donde no se podía descargar la información, porque la página en donde se debía descargar, no estaba disponible. Finalmente, se obtuvo vínculos actualizados por parte de los encargados de Apache Rampart.

5.4. Resumen

En este capítulo se ha detallado la implementación que fue descrita en el capítulo 4, la cual trata la integración de la seguridad en XPDL/BPEL desde un Proceso de Negocio Seguro. En la sección 5.1 se analizó la integración de la seguridad en XPDL, para ello, se utilizaron las propiedades ya existentes en el lenguaje, las cuales son los atributos extendidos. Éstos cumplen el rol de enriquecer el diagrama con información adicional, por lo tanto, es aquí donde se detalla la integración de la seguridad. En el ejemplo ilustrativo de la Figura 5.1, se agregó un atributo extendido a la actividad Rellenar información del paciente, permitiéndole que el código XPDL considere que esta actividad posee un Control de Acceso. Es la forma en que se especifica la seguridad en XPDL. De manera que, cuando se traspase el diagrama BPMN seguro entre herramientas, el código XPDL contendrá la especificación de seguridad.

En la sección 5.2 se presentó la integración de la seguridad en BPEL la cual involucró un trabajo mayor. Para crear un BPMN seguro, fue necesario crear un plugin que permitiera especificar requerimientos de seguridad al diagrama BPMN. A partir de esta especificación, es posible obtener el código BPEL desde donde se desarrolla el código WSDL agregando las políticas o estándares de WS-Security donde finalmente, obtener un Servicio Web seguro. Esta propuesta tuvo principalmente dificultades técnicas, ya que algunas tecnologías (BPMN2BPEL) que son la base para la integración de la seguridad en BPEL, están limitadas a ciertas versiones que no necesariamente funcionan con todas las herramientas descritas en la Figura 5.5.

Capítulo 6

Conclusiones

En esta tesis de magíster se ha propuesto la integración de la seguridad en XPD/L/BPEL desde Procesos de Negocio Seguro, basado en el requerimiento de seguridad de Access Control. Para ello, se usaron tecnologías y estándares que permiten integrar y especificar aspectos y requerimientos de seguridad en lenguajes emergentes que permiten representar el diagrama BPMN.

El objetivo general de esta tesis de magíster quedó expresado en el Capítulo 1. el cual es:

Integrar requerimientos de seguridad en XPD/L/BPEL a partir de Procesos de Negocio Seguro

A continuación, se detallará la valoración de los objetivos específicos que se derivaron del objetivo principal.

- Integrar la seguridad en XPD/L con el objetivo de que, desde un diagrama BPMN seguro, se pueda traducir a código XPD/L abarcando requerimientos de seguridad.

Se ha llevado a cabo un estudio sobre las tecnologías que permiten el desarrollo e integración de la seguridad en el lenguaje XPD/L. Del conjunto de herramientas investigadas se ha elegido TIBCO, ya que es una herramienta open source y que está basada en Eclipse Ganymede, lo que permite la extensión de otras propiedades. La investigación radicó en que la seguridad en XPD/L se puede agregar usando propiedades ya existentes en el lenguaje. XPD/L tiene como objetivo almacenar y transportar la información de negocio que es capaz de describir la notación BPMN. Pero este lenguaje no requiere de un motor de ejecución para su funcionamiento, esto quiere decir, que el lenguaje no almacena funciones que requieren un proceso. En otras palabras, XPD/L solo guarda la información de un diagrama, pero no lo ejecuta.

Por lo anterior, si XPD/L es capaz de soportar algún requerimiento de integración de seguridad, es posible por los atributos extendidos, propiedad del lenguaje XPD/L. Los atributos extendidos son extensiones que complementan el diagrama que se está modelando. Es por eso que estos atributos son capaces de almacenar la referencia de seguridad. Por

ejemplo, si se tiene un diagrama BPMN donde el analista de negocio determina que en una actividad existe Access Control, XPDL sólo guarda la referencia del atributo extendido que especifica que en esa actividad está tal requerimiento de seguridad, pero XPDL no es capaz de soportar algún mecanismo que verifique un control de acceso.

Es así, que la integración de la seguridad en XPDL sí es posible mediante los atributos extendidos. Cabe destacar que esta tesis se enfocó en la integración de Access Control a XPDL, pero la investigación reflejó que se pueden referenciar todos los requerimientos de seguridad. De esta manera, estos requerimientos serán considerados al momento en que el diagrama BPMN seguro sea trasladado entre herramientas.

- Integrar la seguridad en BPEL desde un diagrama BPMN seguro, con el objetivo de crear Servicios Web seguros.

Se desarrolló un estudio sobre los trabajos relacionados con la integración de la seguridad en BPEL desde Procesos de Negocio Seguros. La literatura mostró que el tema de seguridad en BPEL ha sido tratado por varios autores, pero los trabajos que hablan exclusivamente de la seguridad en BPEL son escasos. Estos trabajos proponen entornos ajenos al lenguaje para posteriormente conectarlos con BPEL, tales como controles de acceso y no repudio. Ambos mecanismos de seguridad son integrados a BPEL, pero no se derivan de los Procesos de Negocio Seguros.

Para lograr la integración de la seguridad en BPEL se hizo un estudio de las herramientas que son compatibles con la propuesta de integración. Este estudio mostró que la seguridad en BPEL es limitada por la capacidad del motor de ejecución en donde BPEL trabaja, Apache ODE, por lo tanto, la investigación se enfocó en los archivos WSDL. Este tipo de archivos nacen de un archivo BPEL y su función es almacenar reglas y funciones que serán llevadas a un Servicio Web, luego, es aquí donde sí se permite la seguridad. Para conectar la seguridad entre los archivos BPEL y WSDL, se crean las variables (que nacen de un BPMN seguro) que serán utilizadas en el Servicio Web en el código BPEL, ya que, una vez que se genera el código WSDL desde el código BPEL, el archivo WSDL contendrá las variables que fueron definidas en BPEL. Una vez obtenido lo anterior, se agregan las políticas de seguridad en el código WSDL y se obtiene un Servicio Web seguro con requerimientos de seguridad obtenidos desde un BPMN seguro. Cabe destacar que cada requerimiento de seguridad definido en [68] tiene su similar (algunos de forma parcial) con los estándares de WS-Security v Policy.

A partir de los puntos anteriores, se ha podido establecer de que la creación de Servicios Web seguros desde un SBP es posible. Para probar lo anterior, se hizo un experimento con el requerimiento de Access Control que está definido en un SBP. La experimentación realizada en esta tesis mostró que si es posible y que se puede seguir mejorando. A su vez, se ha realizado una tabla comparativa en que se ha asociado cada requerimiento de seguridad con algún estándar WS-Security o trabajo relacionado (ver Tabla 4.1). Esta Tabla no es definitiva, ya que se han asociado

los requerimientos de SBP y WS-Security a partir de la similitud de sus especificaciones, pero pueden haber trabajos más completos sobre seguridad en Servicios Web y que se pueden relacionar de mejor manera con los requerimientos de seguridad definidos en un SBP. De esta manera, como trabajo futuro quedará implementar todos los requerimientos de seguridad y reflejarlos en el modelador BPMN de Eclipse Indigo.

6.1. Principales aportes

En esta tesis de maestría, los principales aportes fueron:

- Establecer un camino para transformar de manera semi-automática un BPMN seguro en un Servicio Web, usando el lenguaje BPEL, considerando el uso de los estándares WS-Security y Policy.
- Creación de un plugin para BPMN 2.0 modeler Eclipse Indigo con el objetivo de que considere mecanismos de seguridad basados en la especificación de BPMN versión 2.0.

6.2. Contraste de resultados

A continuación, se presentarán los trabajos que fueron presentados en diferentes congresos.

- Márquez G., Rodríguez A., Fernández-Medina E., Revisión de la literatura sobre integración de especificaciones de seguridad en BPEL/XPDL desde Procesos de Negocio Seguros. II Congreso Internacional de Computación e Informática del Norte (INFONOR). 2011 [49].
- Márquez G., Rodríguez A., Fernández-Medina E., Hacia la integración de la seguridad en BPEL desde Procesos de Negocio Seguro. Encuentro Chileno de Computación (ECC). Jornadas Chilenas de Computación. 2011 [47].
- Márquez G., Rodríguez A., Fernández-Medina E., Integración de especificaciones de seguridad en BPEL/XPDL desde Procesos de Negocio Seguro. Encuentro Tesistas (ET). Jornadas Chilenas de Computación. 2011 [48].

Bibliografía

- [1] van der Aalst, W.M.: Patterns and XPD L: A Critical Evaluation of the XML Process Definition Language. BPMcenter.org (2003)
- [2] van der Aalst, W., Dumas, M., ter Hofstede, A., Russell, N., Verbeek, H., Wohed, P.: Life After Bpel? International Workshop on Web Services and Formal Methods (WS-FM) (2005)
- [3] van der Aalst, W., ter Hofstede, A., Weske, M.: Business Process Management: A survey. BPM Center Report (2) (2003)
- [4] Apache: OMXMLSecurity. <https://issues.apache.org/jira/browse/RAMPARTC/component/12311702>
- [5] Apache: Apache Axis2. <http://axis.apache.org/axis2/java/core/> (2011)
- [6] Apache: Apache ODE. <http://ode.apache.org/> (2011)
- [7] Apache: Apache Rampart. <http://axis.apache.org/axis2/java/rampart/> (2011)
- [8] Apache: Apache Tomcat 6. <http://tomcat.apache.org/> (2011)
- [9] Bertino, E., Crampton, J., Paci, F.: Access Control and Authorization Constraints for WS-BPEL. ICWS pp. 275–284 (2006)
- [10] Bilal, M., Thomas, J.P., Harrington, P., Abraham, A.: BPEL Processes for Non-Repudiation Protocols in Web Services. NWeSP 1. 337–340 (2005)
- [11] Biolchini, J., Gomes, P., Candida, A., Horta, G.: Systematic Review in Software Engineering. Springer Berlin Heidelberg 679. 165–176 (2005)
- [12] Caro, A., Rodríguez, A., Calero, C., Fernández-Medina, E., Piattini, M.: Análisis y revisión de la literatura en el contexto de proyectos de fin de carrera: una propuesta (2008)
- [13] Chaolin, Z., Quanwang, S.: Research on Comparative Analysis of BPEL and XPD L. Computer Network (2010)
- [14] Charfi, A., Mezini, M.: Using Aspects for Security Engineering of Web Service Compositions. ICWS 1. 59 – 66 (2005)
- [15] Coalition, W.M.: XPD L. <http://www.xpdl.org/> (2011)

- [16] Cook, M.A.: Building enterprise information architectures : reengineering information systems. Hewlett-Packard professional books. Upper Saddle River, NJ. Prentice Hall. (1996)
- [17] Davenport, T.: Process Innovation: Reengineering work trough information technology. Harvard Business School Press (1993)
- [18] Dikmans, L.: Transforming BPMN into BPEL: Why and How. Tech. rep., Oracle Technology Network (2008)
- [19] Eclipse: Eclipse Indiqo. <http://eclipse.org/indiqo/>
- [20] Eclipse: Eclipse BPEL Editor. <http://www.eclipse.org/bpel/> (2011)
- [21] Eclipse: Eclipse BPMN Modeler. <http://www.eclipse.org/bpmn/> (2011)
- [22] Eclipse: Eclipse Ganymede. <http://www.eclipse.org/qanymede/> (2011)
- [23] Eclipse: Eclipse Web Tool Platform. <http://www.eclipse.org/webtools> (2011)
- [24] Frankova, G.: Engineering Business Process with Service Level Agreements. Ph.D. thesis, International Doctorate School in Information and Communication Technologies (2010)
- [25] García-Bañuelos, L.: Pattern identification and classification in the translation from bpmn to bpel. Proceedings of the OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008. Part I on On the Move to Meaningful Internet Systems pp. 436–444 (2008)
- [26] Grefen, P.: Implementation and integration of inter-organizational processes in a SOA architecture. International Workshop BPM III (2011)
- [27] Gruschka, N., Luttenberger, N.: Protecting Web Services from DoS Attacks by SOAP Message Validation. in Proceedings of the IFIP TC11 21 International Information Security Conference SEC (2006)
- [28] Gutiérrez, C., Rosado, D.G., Fernández-Medina, E.: The practical application of a process for eliciting and designing security in web service systems. Information and Software Technology 51. 1712–1738 (2009)
- [29] Hummer, M., Champy, J.: Reengineering the Corporation: A Manifesto for Business Revolution. Harper Business (1993)
- [30] Jung, J.: Mapping of business process models to workflow schemata-An example using MEMO-OrgML and XPD. Arbeitsberichte des Instituts für Wirtschaftsinformatik (47) (2004)
- [31] Juric, M.: WSDL and BPEL extensions for Event Driven Architecture. Information and Software Technology (52). 1023–1043 (2010)

- [32] Jürjens, J.: Modelling audit security for smart-card payment schemes with UMLSec. SEC pp. 93–108 (2001)
- [33] Jürjens, J.: Secure Java development with UML. Network Security pp. 107–124 (2001)
- [34] Jürjens, J.: Towards Delevelopment of Secure Systems using UMLSec. FASE pp. 187–200 (2001)
- [35] Jürjens, J.: Using UMLSec and Goal Trees for Secure Systems Development. SAC pp. 1026–1030 (2002)
- [36] Jürjens, J.: Developing Safety-Critical Systems with UML. UML pp. 360–372 (2003)
- [37] Jürjens, J.: Secure Systems Development with UML. Springer-Verlag (2005)
- [38] Jürjens, J., Shabalin, P.: Tools for Secure Systems Development with UML: Security Analysis with ATPs. FASE pp. 305–309 (2005)
- [39] Kamoun, F.: A Roadmap towards the Convergence of Business Process Management and Service Oriented Architecture. Ubiquity 2007 (2007)
- [40] Kitchenham, B.: Guideline for performing Systematic Literature Review in Software Engineering. Software Engineering Group, Department of Computer Science, Keele University 2.7 (2007)
- [41] Ko, R., Lee, S., Lee, E.W.: Business process management (BPM) standars: a survey. Emerald 15(5). 744–791 (2009)
- [42] Koning, M., ai Sun, C., Sinnema, M., Avgeriou, P.: VxBPEL: Supporting variability for Web services in BPEL. Inf. Softw. Technol. 51(2). 258–269 (2009)
- [43] van Lessen, T.: Business Process Management with BPMN & BPEL. International Workshop BPM III (2011)
- [44] Li, Y.H., Paik, H.Y., Benatallah, B., Benbernou, S.: Formal Consistency Verification between BPEL Process and Privacy Policy. In Privacy Security and Trust 212-223 (2006)
- [45] Lopez, J., Montenegro, J.A., Vivas, J.L., Okamoto, E., Dawson, E.: Specification and design of advanced authentication and authorization services. Computer Standards and Interfaces (27). 467–478 (2005)
- [46] Lu, Y., Zhang, L., Sun, J.: Task-activity based access control for process collaboration environments. Computers in Industry 60(403-415) (2009)
- [47] Márquez, G., Rodríguez, A., Fernández-Medina, E.: Hacia la integración de la seguridad en BPEL desde Procesos de Negocio Seguro. Encuentro Chileno de Computación (ECC). Jornadas Chilenas de Computación (2011)

- [48] Márquez, G., Rodríguez, A., Fernández-Medina, E.: Integración de especificaciones de seguridad en BPEL/XPDL desde Procesos de Negocio Seguro. Encuentro Tesista (ET). Jornadas Chilenas de Computación (2011)
- [49] Márquez, G., Rodríguez, A., Fernández-Medina, E.: Revisión de la literatura sobre integración de especificaciones de seguridad en BPEL/XPDL desde Procesos de Negocio Seguro. Congreso Internacional de Computación e Informática del Norte de Chile INFONOR-CHILE (2011)
- [50] Menzel, M., Thomas, I., Meinel, C.: Security Requirements Specification in Service-oriented Business Process Management. International Conference on Availability, Reliability and Security (2009)
- [51] Meziane, H., Benbernou, S.: A dynamic privacy model for web services. Computer Standards and Interfaces 32 (2010)
- [52] zur Muehlen, M.: Workflow-based Process Controlling. Advances in Information Systems and Management Science (2002)
- [53] Nassar, P.B., Badr, Y., Biennier, F., Barbar, K.: Extended BPEL with Heterogeneous Authentication Mechanisms in Service Ecosystems. Proceedings of the International Conference on Management of Emergent Digital Ecosystems pp. 126–133 (2009)
- [54] Network, T.D.: TIBCO Business Studio. <http://developer.tibco.com/businessstudio/default.jsp> (2011)
- [55] OASIS: Web Services Security Policy Language (WS-SecurityPolicy) (2005)
- [56] OASIS: Web Services Security UsernameToken Profile 1.1. Tech. rep., OASIS Standard Specification (2006)
- [57] OASIS: Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of XACML v2.0 for Healthcare (1.0) (2009)
- [58] OASIS: OASIS Open stAndarS for the Information Society. <http://www.oasis-open.org/> (2011)
- [59] OASIS: Web Service Security. <http://www.oasis-open.org/committees/wss> (2011)
- [60] OMG: Business Modeling Model and Notation. <http://www.bpmn.org/> (2011)
- [61] OpenSSL: The openssl projec. <http://www.openssl.org/> (2009)
- [62] Ouyang, C., Dumas, M., Breutel, S., ter Hofstede, A.H.M.: Translating Standard Process Models to BPEL. CAiSE (2006)

- [63] Ouyang, C., Dumas, M., Hofstede, A.H.M.T.: Translating BPMN to BPEL. BPM Center Report (2006)
- [64] Ouyang, C., Dumas, M., Hofstede, A.H.M.T., van der Aalst, W.M.: From BPMN Process Models to BPEL Web Services. Proceedings of the IEEE International Conference on Web Services (2006)
- [65] Paci, F., Bertino, E., Crampton, J.: An Access-Control Framework for WS-BPEL. Journal of Web Services Research 5(3). 20–43 (2008)
- [66] Planning, C., Business, S.: The BPMN-XPDL-BPEL value chain. <http://social-biz.org/2006/05/26/bpmn-xpdl-and-bpel/> (2006)
- [67] Reactivity: Web Services Security: Non-repudiation. xml.coverpages.org/ReactivityWS-NonRepudiation-05.pdf (2003)
- [68] Rodríguez, A., Fernández-Medina, E., Piattini, M.: A BPMN Extension for the Modeling of Security Requirements in Business Processes. IEICE transf. Information and Systems E90-D(4). 745–752 (2007)
- [69] Rodríguez, A., Fernández-Medina, E., Trujillo, J., Piattini, M.: Secure Business Process Model specification through a UML 2.0 Activity Diagram Profile. Decision Support Systems 51, 446–465 (2011)
- [70] Rudolph, C., Kuntze, N., Velikova, Z.: Secure Web Service Workflow Execution. ENTCS 236 (2009)
- [71] Shapiro, R.: A Technical Comparison of XPDL, BPML and BPEL4WS. Cape Visions (2002)
- [72] Shapiro, R.: XPDL 2.2: Incorporating BPMN 2.0 Process Modeling Extensions (2010)
- [73] Sindre, G.: Mal-Activity Diagrams for Capturing Attacks on Business Processes. Lecture Notes in Computer Science 4542. 355–366 (2007)
- [74] Sindre, G., Opdahl, A.: Capturing Security Requirements through Misuse Cases (2001)
- [75] Sindre, G., Opdahl, A.: Eliciting security requirements with misuse cases. Requirements Eng. pp. 34–44 (2005)
- [76] Sindre, G., Opdahl, A.: ReqSec: Requirements for Secure Information Systems. FRITEK (2007)
- [77] Souza, A.R., Silva, B.L., Lins, F.A., Damasceno, J.C., Rosa, N.S., Maciel, P.R., Medeiros, R.W., Stephenson, B., Motahari-Nezhad, H.R., Li, J., Northfleet, C.: Incorporating Security Requirements into Service Composition: From Modelling to Execution. Service-Oriented Computing 5900. 373–388 (2009)

- [78] Standard, O.: Web Services Business Process Execution Language Version 2.0. OASIS (2007)
- [79] Standard, W.M.C.W.: Process Definition Interface–XML Process Definition Language 2.1a (2008)
- [80] Together: Together XPD L. <http://www.together.at/prod/workflow/twe> (2011)
- [81] UML: UML resource page. www.uml.org (2011)
- [82] w3.org: Extensible markup language (xml). <http://www.w3.org/XML>
- [83] w3.org: Web services description language (wsdl) 1.1. <http://www.w3.org/TR/wsdl>
- [84] w3.org: Xml path language (xpath) version 1.0. <http://www.w3.org/TR/xpath/>
- [85] w3.org: Xsl transformations (xslt) version 1.0. <http://www.w3.org/TR/xslt>
- [86] w3.org: XML Schema Definition. <http://www.w3.org/XML/Schema> (2011)
- [87] WfCM: Workflow Management Coalition: Terminology & Glossary p. 10 (1999)
- [88] White, S.: XPD L and BPMN (2003)
- [89] Wolter, C., Menzel, M., Meinel, C.: Modelling Security Goals in Business Processes. Computer and Information Science (2008)
- [90] Wolter, C., Menzel, M., Schaad, A., Miseldine, P., Meinel, C.: Model-driven business process security requirement specification. Journal of Systems Architecture (55). 211–223 (2009)
- [91] Yuan, P., Jin, H., Yuan, S., Cao, W., Jiang, L.: WFTXB: A Tool for Translating Between XPD L and BPEL. The 10th IEEE International Conference on High Performance Computing and Communications pp. 647–652 (2008)

Apéndice A

Anexo: Código XPDL del ejemplo ilustrativo

Código XPDL de la Figura 5.1 del Capítulo 5.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xpdI2:Package xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3 xmlns:database="http://www.tibco.com/XPDL/database1.0.0"
4 xmlns:ejjava="http://www.tibco.com/XPDL/EJJava1.0.0"
5 xmlns:email="http://www.tibco.com/XPDL/email1.0.0"
6 xmlns:iProcessExt="http://www.tibco.com/XPDL/iProcessExt1.0.0"
7 xmlns:orchestrator="http://www.tibco.com/XPDL/orchestrator1.0.0"
8 xmlns:order="http://www.tibco.com/XPDL/order1.0.0"
9 xmlns:simulation="http://www.tibco.com/XPDL/Simulation1.0.1"
10 xmlns:xpdExt="http://www.tibco.com/XPDL/xpdExtension1.0.0"
11 xmlns:xpdI2="http://www.wfmc.org/2008/XPDL2.1"
12 xsi:schemaLocation="http://www.wfmc.org/2008/XPDL2.1
13 http://www.wfmc.org/standards/bpmnXpdl_31.xsd"
14 Id="_pcRnYeWIEeC0_c0iFHcd4Q"
15 xpdExt:DisplayName="ProcessPackage" Name="ProcessPackage">
16 <xpdI2:PackageHeader xpdExt:Language="es_CL">
17 <xpdI2:XPDLVersion>2.1</xpdI2:XPDLVersion>
18 <xpdI2:Vendor>TIBCO</xpdI2:Vendor>
19 <xpdI2:Created>2011-09-22</xpdI2:Created>
20 <xpdI2:Description></xpdI2:Description>
21 <xpdI2:Documentation></xpdI2:Documentation>
22 <xpdI2:CostUnit>CLP</xpdI2:CostUnit>
23 </xpdI2:PackageHeader>
```



```

24 <xpd12:RedefinableHeader PublicationStatus="UNDER_REVISION">
25 <xpd12:Author>Gastón</xpd12:Author>
26 <xpd12:Version>1.0.0.qualifier</xpd12:Version>
27 </xpd12:RedefinableHeader>
28 <xpd12:Pools>
29 <xpd12:Pool Id="_rwxZkOWIEeCO_c0iFHcd4Q"
30 xpdExt:DisplayName="CentroAtencionSalud"
31 Name="CentroAtencionSalud"
32 BoundaryVisible="true" Process="_pcRnYOWIEeCO_c0iFHcd4Q">
33 <xpd12:Lanes>
34 <xpd12:Lane Id="_SQRlYOWJIEeCO_c0iFHcd4Q"
35 xpdExt:DisplayName="Admisión" Name="Admisin">
36 <xpd12:NodeGraphicsInfos>
37 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
38 FillColor="128,155,183" Height="303.0" IsVisible="true"/>
39 </xpd12:NodeGraphicsInfos>
40 </xpd12:Lane>
41 <xpd12:Lane Id="_dBknsBIEeGQlu4Khz_9KQ"
42 xpdExt:DisplayName="Evaluación médica"
43 Name="Evaluacinmdica">
44 <xpd12:NodeGraphicsInfos>
45 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
46 FillColor="128,155,183" Height="300.0" IsVisible="true"/>
47 </xpd12:NodeGraphicsInfos>
48 </xpd12:Lane>
49 <xpd12:Lane Id="_fqB3sBIEeGQlu4Khz_9KQ"
50 xpdExt:DisplayName="Contabilidad" Name="Contabilidad">
51 <xpd12:NodeGraphicsInfos>
52 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
53 FillColor="128,155,183" Height="300.0" IsVisible="true"/>
54 </xpd12:NodeGraphicsInfos>
55 </xpd12:Lane>
56 </xpd12:Lanes>
57 <xpd12:NodeGraphicsInfos>
58 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
59 FillColor="223,179,0" IsVisible="true"/>
60 </xpd12:NodeGraphicsInfos>
61 </xpd12:Pool>
62 </xpd12:Pools>

```

```

63 <xpd12:Artifacts>
64 <xpd12:Artifact Id="_ypsfUBlhEeGQlu4Khz_9KQ"
65 Name="Información social" ArtifactType="Group">
66 <xpd12:NodeGraphicsInfos>
67 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
68 Height="268.0"
69 LaneId="_pcRnYOWIEeCO_c0iFHcd4Q"
70 Width="725.0">
71 <xpd12:Coordinates XCoordinate="1402.0"
72 YCoordinate="640.0"/>
73 </xpd12:NodeGraphicsInfo>
74 </xpd12:NodeGraphicsInfos>
75 </xpd12:Artifact>
76 </xpd12:Artifacts>
77 <xpd12:WorkflowProcesses>
78 <xpd12:WorkflowProcess Id="_pcRnYOWIEeCO_c0iFHcd4Q"
79 xpdExt:DisplayName="ProcesoDePacienteDiagram"
80 Name="ProcesoDePacienteDiagram">
81 <xpd12:ProcessHeader>
82 <xpd12:Description></xpd12:Description>
83 </xpd12:ProcessHeader>
84 <xpd12:Activities>
85 <xpd12:Activity Id="_o28TMBleEeGQlu4Khz_9KQ"
86 Name="INICIO"
87 xpdExt:Visibility="Private" xpdExt:DisplayName="INICIO">
88 <xpd12:Event>
89 <xpd12:StartEvent Trigger="None"/>
90 </xpd12:Event>
91 <xpd12:NodeGraphicsInfos>
92 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
93 FillColor="255,219,74" Height="27.0"
94 LaneId="_SQRlYOWJEEeCO_c0iFHcd4Q" Width="27.0">
95 <xpd12:Coordinates XCoordinate="52.0"
96 YCoordinate="136.0"/>
97 </xpd12:NodeGraphicsInfo>
98 </xpd12:NodeGraphicsInfos>
99 </xpd12:Activity>
100 <xpd12:Activity Id="_xnsXlBleEeGQlu4Khz_9KQ"
101 Name="Recibir solicitud admisinpaciente">

```

```

102 xpdExt:Visibility="Private" xpdExt:Display
103 Name="Recibir solicitud admisión paciente">
104 <xpdI2:Implementation>
105 <xpdI2:No/>
106 </xpdI2:Implementation>
107 <xpdI2:NodeGraphicsInfos>
108 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
109 FillColor="255,219,74" Height="64.0"
110 LaneId="_SQRlYOWJEEc0_c0iFHcd4Q" Width="96.0">
111 <xpdI2:Coordinates XCoordinate="160.0"
112 YCoordinate="137.0"/>
113 </xpdI2:NodeGraphicsInfo>
114 </xpdI2:NodeGraphicsInfos>
115 </xpdI2:Activity>
116 <xpdI2:Activity Id="_70kx0BleEeGQlu4Khz_9KQ"
117 Name="Revisarsolicitudpaciente"
118 xpdExt:Visibility="Private" xpdExt:Display
119 Name="Revisar solicitud paciente">
120 <xpdI2:Implementation>
121 <xpdI2:No/>
122 </xpdI2:Implementation>
123 <xpdI2:NodeGraphicsInfos>
124 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
125 FillColor="255,219,74" Height="64.0"
126 LaneId="_SQRlYOWJEEc0_c0iFHcd4Q" Width="96.0">
127 <xpdI2:Coordinates XCoordinate="300.0"
128 YCoordinate="137.0"/>
129 </xpdI2:NodeGraphicsInfo>
130 </xpdI2:NodeGraphicsInfos>
131 </xpdI2:Activity>
132 <xpdI2:Activity Id="_-k9e0BleEeGQlu4Khz_9KQ"
133 Name="Iniciorecoleccindedatos"
134 xpdExt:DisplayName="Inicio recolección de datos">
135 <xpdI2:Route GatewayType="Exclusive"
136 MarkerVisible="true" ExclusiveType="Data"/>
137 <xpdI2:TransitionRestrictions>
138 <xpdI2:TransitionRestriction>
139 <xpdI2:Split Type="Exclusive" ExclusiveType="Data">
140 <xpdI2:TransitionRefs>

```

```

141 <xpdI2:TransitionRef Id="_D90Z4BIfEeGQlu4Khz_9KQ"/>
142 <xpdI2:TransitionRef Id="_JBsjcBIfEeGQlu4Khz_9KQ"/>
143 </xpdI2:TransitionRefs>
144 </xpdI2:Split>
145 </xpdI2:TransitionRestriction>
146 </xpdI2:TransitionRestrictions>
147 <xpdI2:NodeGraphicsInfos>
148 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
149 FillColor="255,219,74"
150 Height="45.0" LaneId="_SQRlYOWJEeC0_c0iFHcd4Q"
151 Width="43.0">
152 <xpdI2:Coordinates XCoordinate="413.0"
153 YCoordinate="136.0"/>
154 </xpdI2:NodeGraphicsInfo>
155 </xpdI2:NodeGraphicsInfos>
156 </xpdI2:Activity>
157 <xpdI2:Activity Id="_Anbs4BIfEeGQlu4Khz_9KQ"
158 Name="Verificarfichaclinica"
159 xpdExt:Visibility="Private" xpdExt:Display
160 Name="Verificar ficha clínica">
161 <xpdI2:Implementation>
162 <xpdI2:No/>
163 </xpdI2:Implementation>
164 <xpdI2:NodeGraphicsInfos>
165 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
166 FillColor="255,219,74"
167 Height="64.0" LaneId="_SQRlYOWJEeC0_c0iFHcd4Q"
168 Width="96.0">
169 <xpdI2:Coordinates XCoordinate="504.0"
170 YCoordinate="52.0"/>
171 </xpdI2:NodeGraphicsInfo>
172 </xpdI2:NodeGraphicsInfos>
173 </xpdI2:Activity>
174 <xpdI2:Activity Id="_GD25URIfEeGQlu4Khz_9KQ"
175 Name="Capturarinformacindeseuros"
176 xpdExt:Visibility="Private" xpdExt:Display
177 Name="Capturar información de seguros">
178 <xpdI2:Implementation>
179 <xpdI2:No/>

```

```

180 </xpdI2:Implementation>
181 <xpdI2:NodeGraphicsInfos>
182 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
183 FillColor="255,219,74" Height="64.0"
184 LaneId="_fqB3sBleEeGQlu4Khz_9KQ" Width="96.0">
185 <xpdI2:Coordinates XCoordinate="414.0"
186 YCoordinate="164.0"/>
187 </xpdI2:NodeGraphicsInfo>
188 </xpdI2:NodeGraphicsInfos>
189 </xpdI2:Activity>
190 <xpdI2:Activity Id="_QmGwEBlfEeGQlu4Khz_9KQ"
191 Name="Existe" xpdExt:DisplayName="¿Existe?">
192 <xpdI2:Route GatewayType="Exclusive" MarkerVisible="true"
193 ExclusiveType="Data"/>
194 <xpdI2:TransitionRestrictions>
195 <xpdI2:TransitionRestriction>
196 <xpdI2:Split Type="Exclusive" ExclusiveType="Data">
197 <xpdI2:TransitionRefs>
198 <xpdI2:TransitionRef Id="_eu8TABlfEeGQlu4Khz_9KQ"/>
199 <xpdI2:TransitionRef Id="_kpuPYBlfEeGQlu4Khz_9KQ"/>
200 </xpdI2:TransitionRefs>
201 </xpdI2:Split>
202 </xpdI2:TransitionRestriction>
203 </xpdI2:TransitionRestrictions>
204 <xpdI2:NodeGraphicsInfos>
205 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
206 FillColor="255,219,74"
207 Height="61.0" LaneId="_SQRIYOWJEeC0_c0iFHcd4Q"
208 Width="43.0">
209 <xpdI2:Coordinates XCoordinate="631.0"
210 YCoordinate="51.0"/>
211 </xpdI2:NodeGraphicsInfo>
212 </xpdI2:NodeGraphicsInfos>
213 </xpdI2:Activity>
214 <xpdI2:Activity Id="_YpAPgBlfEeGQlu4Khz_9KQ"
215 Name="Crear fichaclnica"
216 xpdExt:Visibility="Private" xpdExt:Display
217 Name="Crear ficha clínica">
218 <xpdI2:Implementation>

```

```

219 <xpd12:No/>
220 </xpd12:Implementation>
221 <xpd12:NodeGraphicsInfos>
222 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
223 FillColor="255,219,74" Height="64.0"
224 LaneId="_SQRlYOWJEEc0_c0iFHcd4Q" Width="96.0">
225 <xpd12:Coordinates XCoordinate="755.0"
226 YCoordinate="52.0"/>
227 </xpd12:NodeGraphicsInfo>
228 </xpd12:NodeGraphicsInfos>
229 </xpd12:Activity>
230 <xpd12:Activity Id="_f8dPABlfEeGQlu4Khz_9KQ"
231 Name="Exmenpreadmisin"
232 xpdExt:Visibility="Private" xpdExt:Display
233 Name="Exámen pre-admisión">
234 <xpd12:Implementation>
235 <xpd12:No/>
236 </xpd12:Implementation>
237 <xpd12:NodeGraphicsInfos>
238 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
239 FillColor="255,219,74"
240 Height="64.0" LaneId="_dBknsBleEeGQlu4Khz_9KQ"
241 Width="96.0">
242 <xpd12:Coordinates XCoordinate="784.0"
243 YCoordinate="99.0"/>
244 </xpd12:NodeGraphicsInfo>
245 </xpd12:NodeGraphicsInfos>
246 </xpd12:Activity>
247 <xpd12:Activity Id="_ovtNUBlfEeGQlu4Khz_9KQ"
248 Name="RelLenarinformacin costos"
249 xpdExt:Visibility="Private" xpdExt:Display
250 Name="RelLenar información costos">
251 <xpd12:Implementation>
252 <xpd12:No/>
253 </xpd12:Implementation>
254 <xpd12:NodeGraphicsInfos>
255 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
256 FillColor="255,219,74"
257 Height="64.0" LaneId="_fqB3sBleEeGQlu4Khz_9KQ"

```

```

258 Width="96.0">
259 <xpdI2:Coordinates XCoordinate="620.0"
260 YCoordinate="164.0"/>
261 </xpdI2:NodeGraphicsInfo>
262 </xpdI2:NodeGraphicsInfos>
263 </xpdI2:Activity>
264 <xpdI2:Activity Id="_uy0mUBlfEeGQlu4Khz_9KQ"
265 Name="Finrecoleccindatos"
266 xpdExt:DisplayName="Fin recolección datos">
267 <xpdI2:Route GatewayType="Exclusive"
268 MarkerVisible="true"
269 ExclusiveType="Data"/>
270 <xpdI2:TransitionRestrictions>
271 <xpdI2:TransitionRestriction>
272 <xpdI2:Join Type="Exclusive" ExclusiveType="Data"/>
273 </xpdI2:TransitionRestriction>
274 </xpdI2:TransitionRestrictions>
275 <xpdI2:NodeGraphicsInfos>
276 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
277 FillColor="255,219,74"
278 Height="45.0" LaneId="_dBknsBleEeGQlu4Khz_9KQ"
279 Width="43.0">
280 <xpdI2:Coordinates XCoordinate="909.0"
281 YCoordinate="98.0"/>
282 </xpdI2:NodeGraphicsInfo>
283 </xpdI2:NodeGraphicsInfos>
284 </xpdI2:Activity>
285 <xpdI2:Activity Id="_8f-54BlfEeGQlu4Khz_9KQ"
286 Name="Evaluacinexmenespaciente"
287 xpdExt:Visibility="Private" xpdExt:Display
288 Name="Evaluación exámenes paciente">
289 <xpdI2:Implementation>
290 <xpdI2:No/>
291 </xpdI2:Implementation>
292 <xpdI2:NodeGraphicsInfos>
293 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
294 FillColor="255,219,74"
295 Height="64.0" LaneId="_dBknsBleEeGQlu4Khz_9KQ"
296 Width="96.0">

```

```

297 <xpd12:Coordinates XCoordinate="1058.0"
298 YCoordinate="99.0"/>
299 </xpd12:NodeGraphicsInfo>
300 </xpd12:NodeGraphicsInfos>
301 </xpd12:Activity>
302 <xpd12:Activity Id="_CYiBoB1gEeGQ1u4Khz_9KQ"
303 Name="Rellenar fichac1nica"
304 xpdExt:Visibility="Private"
305 xpdExt:DisplayName="Rellenar ficha cl1nica">
306 <xpd12:Implementation>
307 <xpd12:No/>
308 </xpd12:Implementation>
309 <xpd12:NodeGraphicsInfos>
310 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
311 FillColor="255,219,74"
312 Height="64.0" LaneId="_dBknsB1eEeGQ1u4Khz_9KQ"
313 Width="96.0">
314 <xpd12:Coordinates XCoordinate="1185.0"
315 YCoordinate="99.0"/>
316 </xpd12:NodeGraphicsInfo>
317 </xpd12:NodeGraphicsInfos>
318 </xpd12:Activity>
319 <xpd12:Activity Id="_GEV1AB1gEeGQ1u4Khz_9KQ"
320 Name="Rellenar informacindelpaciente"
321 xpdExt:Visibility="Private" xpdExt:Display
322 Name="Rellenar informaci3n del paciente">
323 <xpd12:Description>Para esta actividad se
324 necesita un control de acceso, ya que
325 se debe acceder a la informaci3n social
326 del paciente</xpd12:Description>
327 <xpd12:Implementation>
328 <xpd12:No/>
329 </xpd12:Implementation>
330 <xpd12:ExtendedAttributes>
331 <xpd12:ExtendedAttribute Name="AccessControl"
332 Value="1"/>
333 </xpd12:ExtendedAttributes>
334 <xpd12:NodeGraphicsInfos>
335 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"

```



```

336 FillColor="255,219,74"
337 Height="64.0" LaneId="_dBknsBleEeGQlu4Khz_9KQ"
338 Width="96.0">
339 <xpd12:Coordinates XCoordinate="1310.0"
340 YCoordinate="99.0"/>
341 </xpd12:NodeGraphicsInfo>
342 </xpd12:NodeGraphicsInfos>
343 </xpd12:Activity>
344 <xpd12:Activity Id="_NhJ74BleEeGQlu4Khz_9KQ"
345 Name="Existedeuda"
346 xpdExt:DisplayName="¿Existe deuda?">
347 <xpd12:Route GatewayType="Exclusive"
348 MarkerVisible="true"
349 ExclusiveType="Data"/>
350 <xpd12:TransitionRestrictions>
351 <xpd12:TransitionRestriction>
352 <xpd12:Split Type="Exclusive" ExclusiveType="Data">
353 <xpd12:TransitionRefs>
354 <xpd12:TransitionRef Id="_eFdbSbleEeGQlu4Khz_9KQ"/>
355 <xpd12:TransitionRef Id="_ljxnYbleEeGQlu4Khz_9KQ"/>
356 </xpd12:TransitionRefs>
357 </xpd12:Split>
358 </xpd12:TransitionRestriction>
359 </xpd12:TransitionRestrictions>
360 <xpd12:NodeGraphicsInfos>
361 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
362 FillColor="255,219,74"
363 Height="45.0" LaneId="_fqB3SbleEeGQlu4Khz_9KQ"
364 Width="43.0">
365 <xpd12:Coordinates XCoordinate="1404.0"
366 YCoordinate="187.0"/>
367 </xpd12:NodeGraphicsInfo>
368 </xpd12:NodeGraphicsInfos>
369 </xpd12:Activity>
370 <xpd12:Activity Id="_brCnsBleEeGQlu4Khz_9KQ"
371 Name="Pagardeudas"
372 xpdExt:Visibility="Private" xpdExt:Display
373 Name="Pagar deudas">
374 <xpd12:Implementation>

```

```

375 <xpdI2:No/>
376 </xpdI2:Implementation>
377 <xpdI2:NodeGraphicsInfos>
378 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
379 FillColor="255,219,74"
380 Height="67.0" LaneId="_fqB3sBleEeGQlu4Khz_9KQ"
381 Width="102.0">
382 <xpdI2:Coordinates XCoordinate="1572.0"
383 YCoordinate="256.0"/>
384 </xpdI2:NodeGraphicsInfo>
385 </xpdI2:NodeGraphicsInfos>
386 </xpdI2:Activity>
387 <xpdI2:Activity Id="_jCpucBlgEeGQlu4Khz_9KQ"
388 Name="EntregarfichacInica"
389 xpdExt:Visibility="Private" xpdExt:Display
390 Name="Entregar ficha clínica">
391 <xpdI2:Implementation>
392 <xpdI2:No/>
393 </xpdI2:Implementation>
394 <xpdI2:NodeGraphicsInfos>
395 <xpdI2:NodeGraphicsInfo BorderColor="0,0,128"
396 FillColor="255,219,74"
397 Height="64.0" LaneId="_dBknsBleEeGQlu4Khz_9KQ"
398 Width="96.0">
399 <xpdI2:Coordinates XCoordinate="1562.0"
400 YCoordinate="243.0"/>
401 </xpdI2:NodeGraphicsInfo>
402 </xpdI2:NodeGraphicsInfos>
403 </xpdI2:Activity>
404 <xpdI2:Activity Id="_B_uDkBlhEeGQlu4Khz_9KQ"
405 Name="EstrabajadorRiesgosocial"
406 xpdExt:DisplayName="¿Es trabajador? (Riesgo social)">
407 <xpdI2:Route GatewayType="Exclusive" MarkerVisible="true"
408 ExclusiveType="Data"/>
409 <xpdI2:TransitionRestrictions>
410 <xpdI2:TransitionRestriction>
411 <xpdI2:Split Type="Exclusive" ExclusiveType="Data">
412 <xpdI2:TransitionRefs>
413 <xpdI2:TransitionRef Id="_NvePQBlhEeGQlu4Khz_9KQ"/>

```

```

414 <xpd12:TransitionRef Id="_T1BoUB1hEeGQ1u4Khz_9KQ"/>
415 </xpd12:TransitionRefs>
416 </xpd12:Split>
417 </xpd12:TransitionRestriction>
418 </xpd12:TransitionRestrictions>
419 <xpd12:NodeGraphicsInfos>
420 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
421 FillColor="255,219,74" Height="45.0"
422 LaneId="_fqB3sB1eEeGQ1u4Khz_9KQ" Width="43.0">
423 <xpd12:Coordinates XCoordinate="1693.0"
424 YCoordinate="135.0"/>
425 </xpd12:NodeGraphicsInfo>
426 </xpd12:NodeGraphicsInfos>
427 </xpd12:Activity>
428 <xpd12:Activity Id="_1e8JwB1hEeGQ1u4Khz_9KQ"
429 Name="Entregadocumentacinocialfichaclinica"
430 xpdExt:Visibility="Private" xpdExt:Display
431 Name="Entregar documentación social ficha clínica">
432 <xpd12:Implementation>
433 <xpd12:No/>
434 </xpd12:Implementation>
435 <xpd12:NodeGraphicsInfos>
436 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
437 FillColor="255,219,74"
438 Height="64.0" LaneId="_fqB3sB1eEeGQ1u4Khz_9KQ"
439 Width="96.0">
440 <xpd12:Coordinates XCoordinate="1840.0"
441 YCoordinate="83.0"/>
442 </xpd12:NodeGraphicsInfo>
443 </xpd12:NodeGraphicsInfos>
444 </xpd12:Activity>
445 <xpd12:Activity Id="_R1-hsB1hEeGQ1u4Khz_9KQ">
446 <xpd12:Route GatewayType="Exclusive" MarkerVisible="true"
447 ExclusiveType="Data"/>
448 <xpd12:TransitionRestrictions>
449 <xpd12:TransitionRestriction>
450 <xpd12:Join Type="Exclusive" ExclusiveType="Data"/>
451 </xpd12:TransitionRestriction>
452 </xpd12:TransitionRestrictions>

```

```

453 <xpd12:NodeGraphicsInfos>
454 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
455 FillColor="255,219,74"
456 Height="45.0" LaneId="_fqB3sBleEeGQlu4Khz_9KQ"
457 Width="43.0">
458 <xpd12:Coordinates XCoordinate="1989.0"
459 YCoordinate="256.0"/>
460 </xpd12:NodeGraphicsInfo>
461 </xpd12:NodeGraphicsInfos>
462 </xpd12:Activity>
463 <xpd12:Activity Id="_dRa6MBlhEeGQlu4Khz_9KQ"
464 Name="Finverificacindeuda"
465 xpdExt:DisplayName="Fin verificación deuda">
466 <xpd12:Route GatewayType="Exclusive"
467 MarkerVisible="true"
468 ExclusiveType="Data"/>
469 <xpd12:TransitionRestrictions>
470 <xpd12:TransitionRestriction>
471 <xpd12:Join Type="Exclusive"
472 ExclusiveType="Data"/>
473 </xpd12:TransitionRestriction>
474 </xpd12:TransitionRestrictions>
475 <xpd12:NodeGraphicsInfos>
476 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
477 FillColor="255,219,74" Height="76.0"
478 LaneId="_dBknsBleEeGQlu4Khz_9KQ" Width="82.0">
479 <xpd12:Coordinates XCoordinate="2152.0"
480 YCoordinate="242.0"/>
481 </xpd12:NodeGraphicsInfo>
482 </xpd12:NodeGraphicsInfos>
483 </xpd12:Activity>
484 <xpd12:Activity Id="_oBgYoRlhEeGQlu4Khz_9KQ"
485 Name="FIN"
486 xpdExt:Visibility="Private"
487 xpdExt:DisplayName="FIN">
488 <xpd12:Event>
489 <xpd12:EndEvent Result="None"/>
490 </xpd12:Event>
491 <xpd12:NodeGraphicsInfos>

```

```

492 <xpd12:NodeGraphicsInfo BorderColor="0,0,128"
493 FillColor="255,219,74"
494 Height="47.0" LaneId="_dBknsBleEeGQlu4Khz_9KQ"
495 Width="31.0">
496 <xpd12:Coordinates XCoordinate="2246.0"
497 YCoordinate="237.0"/>
498 </xpd12:NodeGraphicsInfo>
499 </xpd12:NodeGraphicsInfos>
500 </xpd12:Activity>
501 </xpd12:Activities>
502 <xpd12:Transitions>
503 <xpd12:Transition Id="_6K4gMBleEeGQlu4Khz_9KQ" Name=""
504 From="_o28TMBleEeGQlu4Khz_9KQ"
505 To="_xnsXlBleEeGQlu4Khz_9KQ">
506 <xpd12:ConnectorGraphicsInfos>
507 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
508 ToolId="XPD.ConnectionInfo"/>
509 </xpd12:ConnectorGraphicsInfos>
510 </xpd12:Transition>
511 <xpd12:Transition Id="_9nqwBleEeGQlu4Khz_9KQ"
512 Name=""
513 From="_xnsXlBleEeGQlu4Khz_9KQ"
514 To="_70kx0BleEeGQlu4Khz_9KQ">
515 <xpd12:ConnectorGraphicsInfos>
516 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
517 ToolId="XPD.ConnectionInfo"/>
518 </xpd12:ConnectorGraphicsInfos>
519 </xpd12:Transition>
520 <xpd12:Transition Id="__02ZYBleEeGQlu4Khz_9KQ"
521 Name=""
522 From="_70kx0BleEeGQlu4Khz_9KQ"
523 To="_-k9e0BleEeGQlu4Khz_9KQ">
524 <xpd12:ConnectorGraphicsInfos>
525 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
526 ToolId="XPD.ConnectionInfo"/>
527 </xpd12:ConnectorGraphicsInfos>
528 </xpd12:Transition>
529 <xpd12:Transition Id="_D90Z4BleEeGQlu4Khz_9KQ" Name=""
530 From="_-k9e0BleEeGQlu4Khz_9KQ"

```

```

531 To="_Anbs4BIfEeGQlu4Khz_9KQ">
532 <xpd12:Condition Type="OTHERWISE"/>
533 <xpd12:ConnectorGraphicsInfos>
534 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
535 ToolId="XPD.ConnectionInfo"/>
536 <xpd12:ConnectorGraphicsInfo
537 ToolId="XPD.EndAnchorPosition">
538 <xpd12:Coordinates XCoordinate=
539 "60.62499999999999" YCoordinate="0.0"/>
540 </xpd12:ConnectorGraphicsInfo>
541 </xpd12:ConnectorGraphicsInfos>
542 </xpd12:Transition>
543 <xpd12:Transition Id="_JBsjcBIfEeGQlu4Khz_9KQ" Name=""
544 From="_-k9e0BIfEeGQlu4Khz_9KQ"
545 To="_GD25URIfEeGQlu4Khz_9KQ">
546 <xpd12:Condition Type="CONDITION">
547 <xpd12:Expression/></xpd12:Condition>
548 <xpd12:ConnectorGraphicsInfos>
549 <xpd12:ConnectorGraphicsInfo BorderColor=
550 "0,0,128" ToolId="XPD.ConnectionInfo"/>
551 </xpd12:ConnectorGraphicsInfos>
552 </xpd12:Transition>
553 <xpd12:Transition Id="_RqrygBIfEeGQlu4Khz_9KQ" Name=""
554 From="_Anbs4BIfEeGQlu4Khz_9KQ"
555 To="_QmGwEBIfEeGQlu4Khz_9KQ">
556 <xpd12:ConnectorGraphicsInfos>
557 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
558 ToolId="XPD.ConnectionInfo"/>
559 </xpd12:ConnectorGraphicsInfos>
560 </xpd12:Transition>
561 <xpd12:Transition Id="_eu8TABIfEeGQlu4Khz_9KQ" Name=""
562 From="_QmGwEBIfEeGQlu4Khz_9KQ"
563 To="_YpAPgBIfEeGQlu4Khz_9KQ">
564 <xpd12:Condition Type="OTHERWISE"/>
565 <xpd12:ConnectorGraphicsInfos>
566 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
567 ToolId="XPD.ConnectionInfo"/>
568 </xpd12:ConnectorGraphicsInfos>
569 </xpd12:Transition>

```

```

570 <xpd12:Transition Id="_kpuPYBIfEeGQlu4Khz_9KQ" Name=""
571 From="_QmGwEBIfEeGQlu4Khz_9KQ"
572 To="_f8dPABIfEeGQlu4Khz_9KQ">
573 <xpd12:Condition Type="CONDITION">
574 <xpd12:Expression/></xpd12:Condition>
575 <xpd12:ConnectorGraphicsInfos>
576 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
577 ToolId="XPD.ConnectionInfo"/>
578 <xpd12:ConnectorGraphicsInfo
579 ToolId="XPD.EndAnchorPosition">
580 <xpd12:Coordinates XCoordinate="58.75"
581 YCoordinate="0.0"/>
582 </xpd12:ConnectorGraphicsInfo>
583 </xpd12:ConnectorGraphicsInfos>
584 </xpd12:Transition>
585 <xpd12:Transition Id="_rvxsEBIfEeGQlu4Khz_9KQ" Name=""
586 From="_GD25URIfEeGQlu4Khz_9KQ"
587 To="_ovtNUBIfEeGQlu4Khz_9KQ">
588 <xpd12:ConnectorGraphicsInfos>
589 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
590 ToolId="XPD.ConnectionInfo"/>
591 </xpd12:ConnectorGraphicsInfos>
592 </xpd12:Transition>
593 <xpd12:Transition Id="_yhvvIBIfEeGQlu4Khz_9KQ" Name=""
594 From="_f8dPABIfEeGQlu4Khz_9KQ"
595 To="_uy0mUBIfEeGQlu4Khz_9KQ">
596 <xpd12:ConnectorGraphicsInfos>
597 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
598 ToolId="XPD.ConnectionInfo"/>
599 </xpd12:ConnectorGraphicsInfos>
600 </xpd12:Transition>
601 <xpd12:Transition Id="_znq1YBIfEeGQlu4Khz_9KQ" Name=""
602 From="_YpAPgBIfEeGQlu4Khz_9KQ"
603 To="_uy0mUBIfEeGQlu4Khz_9KQ">
604 <xpd12:ConnectorGraphicsInfos>
605 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
606 ToolId="XPD.ConnectionInfo"/>
607 <xpd12:ConnectorGraphicsInfo
608 ToolId="XPD.StartAnchorPosition">

```

```

609 <xpd12:Coordinates XCoordinate="9.375"
610 YCoordinate="0.0"/>
611 </xpd12:ConnectorGraphicsInfo>
612 </xpd12:ConnectorGraphicsInfos>
613 </xpd12:Transition>
614 <xpd12:Transition Id="_18TKcBIfEeGQlu4Khz_9KQ" Name=""
615 From="_ovtNUBIfEeGQlu4Khz_9KQ"
616 To="_uy0mUBIfEeGQlu4Khz_9KQ">
617 <xpd12:ConnectorGraphicsInfos>
618 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
619 ToolId="XPD.ConnectionInfo"/>
620 <xpd12:ConnectorGraphicsInfo ToolId="XPD.EndAnchorPosition">
621 <xpd12:Coordinates XCoordinate="50.0"
622 YCoordinate="0.0"/>
623 </xpd12:ConnectorGraphicsInfo>
624 <xpd12:ConnectorGraphicsInfo
625 ToolId="XPD.StartAnchorPosition">
626 <xpd12:Coordinates XCoordinate="10.625"
627 YCoordinate="0.0"/>
628 </xpd12:ConnectorGraphicsInfo>
629 </xpd12:ConnectorGraphicsInfos>
630 </xpd12:Transition>
631 <xpd12:Transition Id="_BcugsBIfEeGQlu4Khz_9KQ"
632 Name=""
633 From="_uy0mUBIfEeGQlu4Khz_9KQ"
634 To="_8f-54BIfEeGQlu4Khz_9KQ">
635 <xpd12:ConnectorGraphicsInfos>
636 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
637 ToolId="XPD.ConnectionInfo"/>
638 </xpd12:ConnectorGraphicsInfos>
639 </xpd12:Transition>
640 <xpd12:Transition Id="_Eo_oYBIfEeGQlu4Khz_9KQ"
641 Name=""
642 From="_8f-54BIfEeGQlu4Khz_9KQ"
643 To="_CYiBoBIfEeGQlu4Khz_9KQ">
644 <xpd12:ConnectorGraphicsInfos>
645 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
646 ToolId="XPD.ConnectionInfo"/>
647 </xpd12:ConnectorGraphicsInfos>

```



```

648 </xpdI2:Transition>
649 <xpdI2:Transition Id="_JMu5UBIlgEeGQlu4Khz_9KQ"
650 Name=""
651 From="_CYiBoBIlgEeGQlu4Khz_9KQ"
652 To="_GEVlABlgEeGQlu4Khz_9KQ">
653 <xpdI2:ConnectorGraphicsInfos>
654 <xpdI2:ConnectorGraphicsInfo BorderColor="0,0,128"
655 ToolId="XPD.ConnectionInfo"/>
656 </xpdI2:ConnectorGraphicsInfos>
657 </xpdI2:Transition>
658 <xpdI2:Transition Id="_UQAZgBIlgEeGQlu4Khz_9KQ"
659 Name=""
660 From="_GEVlABlgEeGQlu4Khz_9KQ"
661 To="_NhJ74BBIlgEeGQlu4Khz_9KQ">
662 <xpdI2:ConnectorGraphicsInfos>
663 <xpdI2:ConnectorGraphicsInfo BorderColor="0,0,128"
664 ToolId="XPD.ConnectionInfo">
665 <xpdI2:Coordinates XCoordinate="13.0"
666 YCoordinate="358.0"/>
667 <xpdI2:Coordinates XCoordinate="-61.0"
668 YCoordinate="0.0"/>
669 </xpdI2:ConnectorGraphicsInfo>
670 <xpdI2:ConnectorGraphicsInfo
671 ToolId="XPD.StartAnchorPosition">
672 <xpdI2:Coordinates XCoordinate="35.0"
673 YCoordinate="0.0"/>
674 </xpdI2:ConnectorGraphicsInfo>
675 <xpdI2:ConnectorGraphicsInfo
676 ToolId="XPD.EndAnchorPosition">
677 <xpdI2:Coordinates XCoordinate="75.0"
678 YCoordinate="0.0"/>
679 </xpdI2:ConnectorGraphicsInfo>
680 </xpdI2:ConnectorGraphicsInfos>
681 </xpdI2:Transition>
682 <xpdI2:Transition Id="_eFdbSBlgEeGQlu4Khz_9KQ"
683 Name=""
684 From="_NhJ74BBIlgEeGQlu4Khz_9KQ"
685 To="_brCnsBIlgEeGQlu4Khz_9KQ">
686 <xpdI2:Condition Type="CONDITION">

```

```

687 <xpd12:Expression/></xpd12:Condition>
688 <xpd12:ConnectorGraphicsInfos>
689 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
690 ToolId="XPD.ConnectionInfo">
691 <xpd12:Coordinates XCoordinate="12.0"
692 YCoordinate="53.0"/>
693 <xpd12:Coordinates XCoordinate="-105.0"
694 YCoordinate="1.0"/>
695 </xpd12:ConnectorGraphicsInfo>
696 <xpd12:ConnectorGraphicsInfo
697 ToolId="XPD.EndAnchorPosition">
698 <xpd12:Coordinates XCoordinate="58.4375"
699 YCoordinate="0.0"/>
700 </xpd12:ConnectorGraphicsInfo>
701 <xpd12:ConnectorGraphicsInfo
702 ToolId="XPD.StartAnchorPosition">
703 <xpd12:Coordinates XCoordinate="50.0"
704 YCoordinate="0.0"/>
705 </xpd12:ConnectorGraphicsInfo>
706 </xpd12:ConnectorGraphicsInfos>
707 </xpd12:Transition>
708 <xpd12:Transition Id="_IjxnYBlgEeGQlu4Khz_9KQ"
709 Name=""
710 From="_NhJ74BlgEeGQlu4Khz_9KQ"
711 To="_jCpucBlgEeGQlu4Khz_9KQ">
712 <xpd12:Condition Type="OTHERWISE"/>
713 <xpd12:ConnectorGraphicsInfos>
714 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
715 ToolId="XPD.ConnectionInfo"/>
716 <xpd12:ConnectorGraphicsInfo
717 ToolId="XPD.StartAnchorPosition">
718 <xpd12:Coordinates XCoordinate="0.0"
719 YCoordinate="0.0"/>
720 </xpd12:ConnectorGraphicsInfo>
721 <xpd12:ConnectorGraphicsInfo
722 ToolId="XPD.EndAnchorPosition">
723 <xpd12:Coordinates XCoordinate="62.18749999999999"
724 YCoordinate="0.0"/>
725 </xpd12:ConnectorGraphicsInfo>

```

```

726 </xpdI2:ConnectorGraphicsInfos>
727 </xpdI2:Transition>
728 <xpdI2:Transition Id="_G0VZEBIhEeGQIu4Khz_9KQ"
729 Name=""
730 From="_brCnsBIgEeGQIu4Khz_9KQ"
731 To="_B_uDkBhEeGQIu4Khz_9KQ">
732 <xpdI2:ConnectorGraphicsInfos>
733 <xpdI2:ConnectorGraphicsInfo BorderColor="0,0,128"
734 ToolId="XPD.ConnectionInfo"/>
735 <xpdI2:ConnectorGraphicsInfo
736 ToolId="XPD.StartAnchorPosition">
737 <xpdI2:Coordinates XCoordinate="85.20710059171599"
738 YCoordinate="0.0"/>
739 </xpdI2:ConnectorGraphicsInfo>
740 </xpdI2:ConnectorGraphicsInfos>
741 </xpdI2:Transition>
742 <xpdI2:Transition Id="_NvePQBhEeGQIu4Khz_9KQ"
743 xpdExt:DisplayName="Si"
744 Name="Si" From="_B_uDkBhEeGQIu4Khz_9KQ"
745 To="_Ie8JwBIhEeGQIu4Khz_9KQ">
746 <xpdI2:Condition Type="OTHERWISE"/>
747 <xpdI2:ConnectorGraphicsInfos>
748 <xpdI2:ConnectorGraphicsInfo BorderColor="0,0,128"
749 ToolId="XPD.ConnectionInfo">
750 <xpdI2:Coordinates XCoordinate="20.0"
751 YCoordinate="-38.0"/>
752 <xpdI2:Coordinates XCoordinate="-79.0"
753 YCoordinate="-4.0"/>
754 </xpdI2:ConnectorGraphicsInfo>
755 <xpdI2:ConnectorGraphicsInfo
756 ToolId="XPD.StartAnchorPosition">
757 <xpdI2:Coordinates XCoordinate="0.0"
758 YCoordinate="0.0"/>
759 </xpdI2:ConnectorGraphicsInfo>
760 <xpdI2:ConnectorGraphicsInfo
761 ToolId="XPD.EndAnchorPosition">
762 <xpdI2:Coordinates XCoordinate="60.9375"
763 YCoordinate="0.0"/>
764 </xpdI2:ConnectorGraphicsInfo>

```

```

765 </xpd12:ConnectorGraphicsInfos>
766 </xpd12:Transition>
767 <xpd12:Transition Id="_T1BoUB1hEeGQ1u4Khz_9KQ"
768 xpdExt:DisplayName="No"
769 Name="No" From="_B_uDkB1hEeGQ1u4Khz_9KQ"
770 To="_R1-hsB1hEeGQ1u4Khz_9KQ">
771 <xpd12:Condition Type="CONDITION">
772 <xpd12:Expression/></xpd12:Condition>
773 <xpd12:ConnectorGraphicsInfos>
774 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
775 ToolId="XPD.ConnectionInfo">
776 <xpd12:Coordinates XCoordinate="43.0"
777 YCoordinate="122.0"/>
778 <xpd12:Coordinates XCoordinate="-253.0"
779 YCoordinate="1.0"/>
780 </xpd12:ConnectorGraphicsInfo>
781 </xpd12:ConnectorGraphicsInfos>
782 </xpd12:Transition>
783 <xpd12:Transition Id="_Wqm-QB1hEeGQ1u4Khz_9KQ"
784 Name="" From="_1e8JwB1hEeGQ1u4Khz_9KQ"
785 To="_R1-hsB1hEeGQ1u4Khz_9KQ">
786 <xpd12:ConnectorGraphicsInfos>
787 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
788 ToolId="XPD.ConnectionInfo">
789 <xpd12:Coordinates XCoordinate="104.0"
790 YCoordinate="0.0"/>
791 <xpd12:Coordinates XCoordinate="-45.0"
792 YCoordinate="-152.0"/>
793 </xpd12:ConnectorGraphicsInfo>
794 <xpd12:ConnectorGraphicsInfo
795 ToolId="XPD.EndAnchorPosition">
796 <xpd12:Coordinates XCoordinate="0.0"
797 YCoordinate="0.0"/>
798 </xpd12:ConnectorGraphicsInfo>
799 </xpd12:ConnectorGraphicsInfos>
800 </xpd12:Transition>
801 <xpd12:Transition Id="_ggXicB1hEeGQ1u4Khz_9KQ"
802 Name="" From="_R1-hsB1hEeGQ1u4Khz_9KQ"
803 To="_dRa6MB1hEeGQ1u4Khz_9KQ">

```

```

804 <xpd12:ConnectorGraphicsInfos>
805 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
806 ToolId="XPD.ConnectionInfo">
807 <xpd12:Coordinates XCoordinate="147.0"
808 YCoordinate="-61.0"/>
809 <xpd12:Coordinates XCoordinate="0.0"
810 YCoordinate="244.0"/>
811 </xpd12:ConnectorGraphicsInfo>
812 <xpd12:ConnectorGraphicsInfo
813 ToolId="XPD.StartAnchorPosition">
814 <xpd12:Coordinates XCoordinate="25.0"
815 YCoordinate="0.0"/>
816 </xpd12:ConnectorGraphicsInfo>
817 <xpd12:ConnectorGraphicsInfo
818 ToolId="XPD.EndAnchorPosition">
819 <xpd12:Coordinates XCoordinate="48.781231420920214"
820 YCoordinate="0.0"/>
821 </xpd12:ConnectorGraphicsInfo>
822 </xpd12:ConnectorGraphicsInfos>
823 </xpd12:Transition>
824 <xpd12:Transition Id="_lYDDEBlhEeGQlu4Khz_9KQ"
825 Name=""
826 From="_jCpucBlgEeGQlu4Khz_9KQ"
827 To="_dRa6MBlhEeGQlu4Khz_9KQ">
828 <xpd12:ConnectorGraphicsInfos>
829 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
830 ToolId="XPD.ConnectionInfo"/>
831 <xpd12:ConnectorGraphicsInfo
832 ToolId="XPD.StartAnchorPosition">
833 <xpd12:Coordinates XCoordinate="10.0"
834 YCoordinate="0.0"/>
835 </xpd12:ConnectorGraphicsInfo>
836 </xpd12:ConnectorGraphicsInfos>
837 </xpd12:Transition>
838 <xpd12:Transition Id="_pomSQBlhEeGQlu4Khz_9KQ"
839 Name=""
840 From="_dRa6MBlhEeGQlu4Khz_9KQ"
841 To="_oBgYoRlhEeGQlu4Khz_9KQ">
842 <xpd12:ConnectorGraphicsInfos>

```

```

843 <xpd12:ConnectorGraphicsInfo BorderColor="0,0,128"
844 ToolId="XPD.ConnectionInfo"/>
845 <xpd12:ConnectorGraphicsInfo
846 ToolId="XPD.StartAnchorPosition">
847 <xpd12:Coordinates XCoordinate="23.16384283197311"
848 YCoordinate="0.0"/>
849 </xpd12:ConnectorGraphicsInfo>
850 </xpd12:ConnectorGraphicsInfos>
851 </xpd12:Transition>
852 </xpd12:Transitions>
853 </xpd12:WorkflowProcess>
854 </xpd12:WorkflowProcesses>
855 <xpd12:ExtendedAttributes>
856 <xpd12:ExtendedAttribute Name="CreatedBy"
857 Value="TIBC0 Business Studio"/>
858 <xpd12:ExtendedAttribute Name="FormatVersion" Value="7"/>
859 </xpd12:ExtendedAttributes>
860 </xpd12:Package>

```

Apéndice B

Anexo: Código BPEL del ejemplo ilustrativo

Código BPEL de la Figura 5.1 del Capítulo 5.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <bpel:process exitOnStandardFault="yes" name="centroAtencionSalud"
3 suppressJoinFailure="yes" targetNamespace="http://tempuri.org/plks"
4 xmlns:bpel=
5 "http://docs.oasis-open.org/wsbpel/2.0/process/executable"
6 xmlns:tns="http://tempuri.org/plks"
7 xmlns:ns1="http://cnti_calidad/procesoDePrestamo"
8 xmlns:ns2="http://cnti_calidad/procesoDeAprobacionPrestamo"
9 xmlns:ns3="http://www.w3.org/2001/XMLSchema">
10 <bpel:import namespace=
11 "http://cnti_calidad/procesoDeAprobacionPrestamo"
12 location="ProcesoDeAprobacionPrestamoArtifacts.wsdl"
13 importType="http://schemas.xmlsoap.org/wsdl/"></bpel:import>
14 <bpel:import namespace="http://cnti_calidad/procesoDePrestamo"
15 location="ProcesoDePrestamoArtifacts.wsdl"
16 importType="http://schemas.xmlsoap.org/wsdl/">
17 </bpel:import>
18 <bpel:partnerLinks>
19 <bpel:partnerLink name="paciente" partnerLinkType=
20 "ns1:ProcesoDePrestamoPLT"
21 myRole="ProcesoDePrestamoProvider"></bpel:partnerLink>
22 <bpel:partnerLink name="entregaFicha" partnerLinkType=
23 "ns2:ProcesoDeAprobacionPrestamoPLT" partnerRole=
```

```

24 "ProcesoDeAprobacionPrestamoProvider"></bpel:partnerLink>
25 </bpel:partnerLinks>
26 <bpel:variables>
27 <bpel:variable name="entrada" messageType=
28 "ns1:ProcesoDePrestamoRequestMessage"></bpel:variable>
29 <bpel:variable name="salida" messageType=
30 "ns1:ProcesoDePrestamoResponseMessage"></bpel:variable>
31 <bpel:variable name="aprobacionFichaRequest" messageType=
32 "ns2:ProcesoDeAprobacionPrestamoRequestMessage"></bpel:variable>
33 <bpel:variable name="aprobacionFichaResponse"
34 messageType="ns2:ProcesoDeAprobacionPrestamoResponseMessage">
35 </bpel:variable>
36 <bpel:variable name="deuda" type="ns3:string"></bpel:variable>
37 </bpel:variables>
38 <bpel:sequence name="INICIO-FIN">
39 <bpel:empty name="INICIO"></bpel:empty>
40 <bpel:empty name="Recibir solicitud admisión paciente"/>
41 <bpel:empty name="Revisar solicitud paciente"/>
42 <bpel:sequence name="null-null">
43 <bpel:if name="Inicio recolección datos-Fin recolección datos">
44 <bpel:sequence name="Verificar ficha clínica-
45 ¿Existe?-_Fin recolección datos_">
46 <bpel:empty name="Verificar ficha clínica"/>
47 <bpel:if name="¿Existe?-_Fin recolección datos_">
48 <bpel:empty name="Crear ficha clínica"/>
49 <bpel:else>
50 <bpel:empty name="Exámen de pre-Admisión"/>
51 </bpel:else>
52 </bpel:if>
53 </bpel:sequence>
54 <bpel:else>
55 <bpel:sequence name="Capturar información de seguros
56 -Rellenar información costos">
57 <bpel:empty name="Capturar información de seguros"/>
58 <bpel:empty name="Rellenar información costos"/>
59 </bpel:sequence>
60 </bpel:else>
61 </bpel:if>
62 </bpel:sequence>

```



```

63 <bpel:empty name="Evaluación exámenes paciente"/>
64 <bpel:empty name="Rellenar ficha clínica"/>
65 <bpel:sequence name="Rellenar información del paciente">
66 <bpel:receive name="InicioPeticion" partnerLink="paciente"
67 operation="peticion" portType="ns1:ProcesoDePrestamoPT"
68 variable="entrada" createInstance="yes"></bpel:receive>
69 <bpel:condition><![CDATA[$entrada.AccessControl]]></bpel:condition>
70 <bpel:assign validate="no" name="Assign"></bpel:assign>
71 <bpel:if name="¿Existe deuda?-Fin verificación deuda">
72 <bpel:condition><![CDATA[$entrada.payload/ns1:Deuda='si']]>
73 </bpel:condition>
74 <bpel:sequence name="Pagar deudas-¿Riesgo Social?-null">
75 <bpel:empty name="Pagar deudas" />
76 <bpel:if name="¿Riesgo Social?-No">
77 <bpel:condition><![CDATA[$entrada.payload/ns1:Empleado_si_no='si']]>
78 </bpel:condition>
79 <bpel:assign validate="no" name="AsignaFichaClinica">
80 <bpel:copy>
81 <bpel:from>
82 <bpel:literal xml:space="preserve"><tns:ProcesoDePrestamoResponse
83 xmlns:tns="http://cnti_calidad/procesoDePrestamo" xmlns:xsi=
84 "http://www.w3.org/2001/XMLSchema-instance">
85 <tns:PrimerNombre></tns:PrimerNombre>
86 <tns:PrimerApellido></tns:PrimerApellido>
87 <tns:MontoPedido></tns:MontoPedido>
88 <tns:Activo></tns:Activo>
89 <tns:Empleado_si_no></tns:Empleado_si_no>
90 <tns:Deuda></tns:Deuda>
91 <tns:RespuestaPrestamoPedido></tns:RespuestaPrestamoPedido>
92 </tns:ProcesoDePrestamoResponse>
93 </bpel:literal>
94 </bpel:from>
95 <bpel:to variable="salida" part="payload"></bpel:to>
96 </bpel:copy>
97 <bpel:copy>
98 <bpel:from part="payload" variable="entrada">
99 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
100 sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]></bpel:query>
101 </bpel:from>

```

```

102 <bpel:to part="payload" variable="salida">
103 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
104 sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]></bpel:query>
105 </bpel:to>
106 </bpel:copy>
107 <bpel:copy>
108 <bpel:from part="payload" variable="entrada">
109 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
110 sublang:xpath1.0">
111 <![CDATA[ns1:PrimerApellido]]>
112 </bpel:query>
113 </bpel:from>
114 <bpel:to part="payload" variable="salida">
115 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
116 sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]></bpel:query>
117 </bpel:to>
118 </bpel:copy>
119 <bpel:copy>
120 <bpel:from part="payload" variable="entrada">
121 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
122 sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]></bpel:query>
123 </bpel:from>
124 <bpel:to part="payload" variable="salida">
125 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
126 sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]></bpel:query>
127 </bpel:to>
128 </bpel:copy>
129 <bpel:copy>
130 <bpel:from part="payload" variable="entrada">
131 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
132 sublang:xpath1.0">
133 <![CDATA[ns1:Deuda]]>
134 </bpel:query>
135 </bpel:from>
136 <bpel:to part="payload" variable="salida">
137 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
138 sublang:xpath1.0"><![CDATA[ns1:Deuda]]></bpel:query>
139 </bpel:to>
140 </bpel:copy>

```

```

141 <bpel:copy>
142 <bpel:from>
143 <bpel:literal xml:space="preserve">Entrega de ficha aprobada
144 </bpel:literal>
145 </bpel:from>
146 <bpel:to part="payload" variable="salida">
147 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
148 sublang:xpath1.0"><![CDATA[ns1:RespuestaPrestamoPedido]]>
149 </bpel:query>
150 </bpel:to>
151 </bpel:copy>
152 </bpel:assign>
153 <bpel:else>
154 <bpel:sequence name="Sequence">
155 <bpel:assign validate="no" name="AsignarDatosAcontabilidad">
156 <bpel:copy>
157 <bpel:from>
158 <bpel:literal xml:space="preserve">
159 <tns:ProcesoDeAprobacionPrestamoRequest xmlns:tns=
160 "http://cnti_calidad/procesoDeAprobacionPrestamo"
161 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
162 <tns:PrimerNombre></tns:PrimerNombre>
163 <tns:PrimerApellido></tns:PrimerApellido>
164 <tns:MontoPedido></tns:MontoPedido>
165 <tns:Activo></tns:Activo>
166 <tns:Empleado_si_no></tns:Empleado_si_no>
167 <tns:Deuda></tns:Deuda>
168 <tns:Notas></tns:Notas>
169 <tns:Login></tns:Login>
170 <tns>Password></tns>Password>
171 </tns:ProcesoDeAprobacionPrestamoRequest>
172 </bpel:literal>
173 </bpel:from>
174 <bpel:to variable="aprobacionFichaRequest"
175 part="payload"></bpel:to>
176 </bpel:copy>
177 <bpel:copy>
178 <bpel:from part="payload" variable="entrada">
179 <bpel:query queryLanguage="urn:oasis:names:tc:

```

```

180 wsbpel:2.0:sublang:xpath1.0">
181 <![CDATA[ns1:PrimerApellido]]>
182 </bpel:query>
183 </bpel:from>
184 <bpel:to part="payload" variable="aprobacionFichaRequest">
185 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang
186 :xpath1.0"><![CDATA[ns2:PrimerNombre]]></bpel:query>
187 </bpel:to>
188 </bpel:copy>
189 <bpel:copy>
190 <bpel:from part="payload" variable="entrada">
191 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
192 sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]></bpel:query>
193 </bpel:from>
194 <bpel:to part="payload" variable="aprobacionFichaRequest">
195 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
196 sublang:xpath1.0"><![CDATA[ns2:PrimerApellido]]></bpel:query>
197 </bpel:to>
198 </bpel:copy>
199 <bpel:copy>
200 <bpel:from part="payload" variable="entrada">
201 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
202 sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]></bpel:query>
203 </bpel:from>
204 <bpel:to part="payload" variable="aprobacionFichaRequest">
205 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
206 sublang:xpath1.0"><![CDATA[ns2:Empleado_si_no]]></bpel:query>
207 </bpel:to>
208 </bpel:copy>
209 <bpel:copy>
210 <bpel:from part="payload" variable="entrada">
211 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
212 sublang:xpath1.0"><![CDATA[ns1:Deuda]]></bpel:query>
213 </bpel:from>
214 <bpel:to part="payload" variable="aprobacionFichaRequest">
215 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
216 sublang:xpath1.0"><![CDATA[ns2:Deuda]]></bpel:query>
217 </bpel:to>
218 </bpel:copy>

```

```

219 </bpel:assign>
220 <bpel:invoke name="ServicioWebDeAprobacion"
221 partnerLink="entregaFicha" operation="aprobacion"
222 portType="ns2:ProcesoDeAprobacionPrestamoPT"
223 inputVariable="aprobacionFichaRequest"
224 outputVariable="aprobacionFichaResponse"></bpel:invoke>
225 <bpel:assign validate="no" name="AsignarFichaRiesgoSocial">
226 <bpel:copy>
227 <bpel:from>
228 <bpel:literal xml:space="preserve">
229 <tns:ProcesoDePrestamoResponse xmlns
230 :tns="http://cnti_calidad/procesoDePrestamo"
231 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
232 <tns:PrimerNombre></tns:PrimerNombre>
233 <tns:PrimerApellido></tns:PrimerApellido>
234 <tns:MontoPedido></tns:MontoPedido>
235 <tns:Activo></tns:Activo>
236 <tns:Empleado_si_no></tns:Empleado_si_no>
237 <tns:Deuda></tns:Deuda>
238 <tns:RespuestaPrestamoPedido></tns:RespuestaPrestamoPedido>
239 </tns:ProcesoDePrestamoResponse>
240 </bpel:literal>
241 </bpel:from>
242 <bpel:to variable="salida" part="payload"></bpel:to>
243 </bpel:copy>
244 <bpel:copy>
245 <bpel:from part="payload" variable="entrada">
246 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
247 sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]></bpel:query>
248 </bpel:from>
249 <bpel:to part="payload" variable="salida">
250 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
251 sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]></bpel:query>
252 </bpel:to>
253 </bpel:copy>
254 <bpel:copy>
255 <bpel:from part="payload" variable="entrada">
256 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
257 sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]></bpel:query>

```

```

258 </bpel:from>
259 <bpel:to part="payload" variable="salida">
260 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
261 :sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]></bpel:query>
262 </bpel:to>
263 </bpel:copy>
264 <bpel:copy>
265 <bpel:from part="payload" variable="entrada">
266 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
267 :sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]></bpel:query>
268 </bpel:from>
269 <bpel:to part="payload" variable="salida">
270 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
271 :sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]></bpel:query>
272 </bpel:to>
273 </bpel:copy>
274 <bpel:copy>
275 <bpel:from part="payload" variable="entrada">
276 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
277 :sublang:xpath1.0">
278 <![CDATA[ns1:Deuda]]>
279 </bpel:query>
280 </bpel:from>
281 <bpel:to part="payload" variable="salida">
282 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
283 :sublang:xpath1.0"><![CDATA[ns1:Deuda]]></bpel:query>
284 </bpel:to>
285 </bpel:copy>
286 <bpel:copy>
287 <bpel:from part="payload" variable="aprobacionFichaResponse">
288 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
289 :sublang:xpath1.0"><![CDATA[ns2:RespuestaPrestamoPedido]]></bpel:query>
290 </bpel:from>
291 <bpel:to part="payload" variable="salida">
292 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0
293 :sublang:xpath1.0"><![CDATA[ns1:RespuestaPrestamoPedido]]></bpel:query>
294 </bpel:to>
295 </bpel:copy>
296 </bpel:assign>

```

```

297 </bpe1:sequence>
298 </bpe1:else>
299 </bpe1:if>
300 </bpe1:sequence>
301 <bpe1:else>
302 <bpe1:sequence name="SequenceFicha">
303 <bpe1:assign validate="no" name="AsignarDatosAcontabilidad">
304 <bpe1:copy>
305 <bpe1:from>
306 <bpe1:literal xml:space="preserve">
307 <tns:ProcesoDeAprobacionPrestamoRequest xmlns:
308 tns="http://cnti_calidad/procesoDeAprobacionPrestamo"
309 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
310 <tns:PrimerNombre></tns:PrimerNombre>
311 <tns:PrimerApellido></tns:PrimerApellido>
312 <tns:MontoPedido></tns:MontoPedido>
313 <tns:Activo></tns:Activo>
314 <tns:Empleado_si_no></tns:Empleado_si_no>
315 <tns:Deuda></tns:Deuda>
316 <tns:Notas></tns:Notas>
317 <tns:Login></tns:Login>
318 <tns>Password></tns>Password>
319 </tns:ProcesoDeAprobacionPrestamoRequest>
320 </bpe1:literal>
321 </bpe1:from>
322 <bpe1:to variable="aprobacionFichaRequest" part="payload">
323 </bpe1:to>
324 </bpe1:copy>
325 <bpe1:copy>
326 <bpe1:from part="payload" variable="entrada">
327 <bpe1:query queryLanguage="urn:oasis:names:tc:wsbpe1:2.0:
328 sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]></bpe1:query>
329 </bpe1:from>
330 <bpe1:to part="payload" variable="aprobacionFichaRequest">
331 <bpe1:query queryLanguage="urn:oasis:names:tc:wsbpe1:2.0:
332 sublang:xpath1.0"><![CDATA[ns2:PrimerNombre]]></bpe1:query>
333 </bpe1:to>
334 </bpe1:copy>
335 <bpe1:copy>

```

```

336 <bpel:from part="payload" variable="entrada">
337 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
338 sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]></bpel:query>
339 </bpel:from>
340 <bpel:to part="payload" variable="aprobacionFichaRequest">
341 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
342 sublang:xpath1.0"><![CDATA[ns2:PrimerApellido]]></bpel:query>
343 </bpel:to>
344 </bpel:copy>
345 <bpel:copy>
346 <bpel:from part="payload" variable="entrada">
347 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
348 sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]></bpel:query>
349 </bpel:from>
350 <bpel:to part="payload" variable="aprobacionFichaRequest">
351 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
352 sublang:xpath1.0"><![CDATA[ns2:Empleado_si_no]]></bpel:query>
353 </bpel:to>
354 </bpel:copy>
355 <bpel:copy>
356 <bpel:from part="payload" variable="entrada">
357 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
358 sublang:xpath1.0"><![CDATA[ns1:Deuda]]></bpel:query>
359 </bpel:from>
360 <bpel:to part="payload" variable="aprobacionFichaRequest">
361 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
362 sublang:xpath1.0"><![CDATA[ns2:Deuda]]></bpel:query>
363 </bpel:to>
364 </bpel:copy>
365 </bpel:assign>
366 <bpel:invoke name="ServicioWebDeAprobacion"
367 partnerLink="entregaFicha" operation="aprobacion"
368 portType="ns2:ProcesoDeAprobacionPrestamoPT"
369 inputVariable="aprobacionFichaRequest"
370 outputVariable="aprobacionFichaResponse"></bpel:invoke>
371 <bpel:assign validate="no" name="AsignarFicha">
372 <bpel:copy>
373 <bpel:from>
374 <bpel:literal xml:space="preserve"><tns:ProcesoDePrestamoResponse

```



```

375 xmlns:tns="http://cnti_calidad/procesoDePrestamo"
376 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
377 <tns:PrimerNombre></tns:PrimerNombre>
378 <tns:PrimerApellido></tns:PrimerApellido>
379 <tns:MontoPedido></tns:MontoPedido>
380 <tns:Activo></tns:Activo>
381 <tns:Empleado_si_no></tns:Empleado_si_no>
382 <tns:Deuda></tns:Deuda>
383 <tns:RespuestaPrestamoPedido></tns:RespuestaPrestamoPedido>
384 </tns:ProcesoDePrestamoResponse>
385 </bpel:literal>
386 </bpel:from>
387 <bpel:to variable="salida" part="payload"></bpel:to>
388 </bpel:copy>
389 <bpel:copy>
390 <bpel:from part="payload" variable="entrada">
391 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
392 sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]>
393 </bpel:query>
394 </bpel:from>
395 <bpel:to part="payload" variable="salida">
396 <bpel:query queryLanguage="urn:oasis:names:tc:
397 wsbpel:2.0:sublang:xpath1.0"><![CDATA[ns1:PrimerNombre]]>
398 </bpel:query>
399 </bpel:to>
400 </bpel:copy>
401 <bpel:copy>
402 <bpel:from part="payload" variable="entrada">
403 <bpel:query queryLanguage="urn:oasis:names:tc:
404 wsbpel:2.0:sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]>
405 </bpel:query>
406 </bpel:from>
407 <bpel:to part="payload" variable="salida">
408 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
409 sublang:xpath1.0"><![CDATA[ns1:PrimerApellido]]>
410 </bpel:query>
411 </bpel:to>
412 </bpel:copy>
413 <bpel:copy>

```

```

414 <bpel:from part="payload" variable="entrada">
415 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
416 sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]>
417 </bpel:query>
418 </bpel:from>
419 <bpel:to part="payload" variable="salida">
420 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
421 sublang:xpath1.0"><![CDATA[ns1:Empleado_si_no]]>
422 </bpel:query>
423 </bpel:to>
424 </bpel:copy>
425 <bpel:copy>
426 <bpel:from part="payload" variable="entrada">
427 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
428 sublang:xpath1.0"><![CDATA[ns1:Deuda]]>
429 </bpel:query>
430 </bpel:from>
431 <bpel:to part="payload" variable="salida">
432 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
433 sublang:xpath1.0"><![CDATA[ns1:Deuda]]>
434 </bpel:query>
435 </bpel:to>
436 </bpel:copy>
437 <bpel:copy>
438 <bpel:from part="payload"
439 variable="aprobacionFichaResponse">
440 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
441 sublang:xpath1.0"><![CDATA[ns2:RespuestaPrestamoPedido]]>
442 </bpel:query>
443 </bpel:from>
444 <bpel:to part="payload" variable="salida">
445 <bpel:query queryLanguage="urn:oasis:names:tc:wsbpel:2.0:
446 sublang:xpath1.0"><![CDATA[ns1:RespuestaPrestamoPedido]]>
447 </bpel:query>
448 </bpel:to>
449 </bpel:copy>
450 </bpel:assign>
451 </bpel:sequence>
452 </bpel:else>

```

```
453 </bpe!l:sequence>
454 </bpe!l:process>
```

Política de seguridad de WSDL para el ejemplo del proceso de centro de atención de salud

```
1 <wsp:Policy wsu:Id="SignOnly" xmlns:wsu=
2   "http://docs.oasis-open.org/wss/2004/01/
3   oasis-200401-wss-wssecurity-utility-1.0.xsd"
4   xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy">
5   <wsp:ExactlyOne>
6     <wsp:All>
7       <sp:AsymmetricBinding xmlns:
8   sp="http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702">
9         <wsp:Policy>
10          <sp:InitiatorToken>
11            <wsp:Policy>
12              <sp:X509Token sp:IncludeToken=
13   "http://docs.oasis-open.org/
14   ws-sx/ws-securitypolicy /200702/IncludeToken/AlwaysToRecipient"/>
15            </wsp:Policy>
16          </sp:InitiatorToken>
17          <sp:RecipientToken>
18            <wsp:Policy>
19              <sp:X509Token sp:IncludeToken=
20   "http://docs.oasis-open.org/
21   ws-sx/ws-securitypolicy/200702/IncludeToken/AlwaysToInitiator"/>
22            </wsp:Policy>
23          </sp:RecipientToken>
24          <sp:AlgorithmSuite>
25            <wsp:Policy>
26              <sp:TripleDesRsa15/>
27            </wsp:Policy>
28          </sp:AlgorithmSuite>
29          <sp:Layout>
30            <wsp:Policy>
31              <sp:Strict/>
32            </wsp:Policy>
```

```

33         </sp:Layout>
34         <sp:IncludeTimestamp/>
35         <sp:OnlySignEntireHeadersAndBody/>
36     </wsp:Policy>
37 </sp:AsymmetricBinding>
38 <sp:SignedParts xmlns:sp="http://docs.oasis-open.org/
39 ws-sx/ws-securitypolicy/200702">
40     <sp:Body/>
41 </sp:SignedParts>
42 </wsp:All>
43 </wsp:ExactlyOne>
44 </wsp:Policy>

```

Política de seguridad de UsernameToken con password asignada para el ejemplo del proceso de centro de atención de salud

```

1 <wsp:Policy wsu:Id="UsernameToken" xmlns:wsu=
2     "http://docs.oasis-open.org/wss/2004/01/oasis-200401-
3 wss-wssecurity-utility-1.0.xsd"
4     xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy">
5     <wsp:ExactlyOne>
6         <wsp:All>
7             <sp:SupportingTokens
8                 xmlns:sp="http://docs.oasis-open.org/ws-sx/
9 ws-securitypolicy/200702">
10                 <wsp:Policy>
11                     <sp:UsernameToken sp:IncludeToken=
12                         "http://docs.oasis-open.org/ws-sx/
13 ws-securitypolicy/200702
14 /IncludeToken/AlwaysToRecipient"/>
15                 </wsp:Policy>
16             </sp:SupportingTokens>
17         </wsp:All>
18     </wsp:ExactlyOne>
19 </wsp:Policy>

```