



**UNIVERSIDAD SIMÓN BOLÍVAR**  
**DECANATO DE ESTUDIOS PROFESIONALES**  
**COORDINACIÓN DE INGENIERÍA EN COMPUTACIÓN**

**DESARROLLO DE PROCESO DE SOLICITUD DE TARJETAS DE CRÉDITO**  
**MEDIANTE ARQUITECTURA SOA-BPM**

Por:

Luis Alejandro Hernández Sousa

**INFORME DE PASANTÍA**

Presentado ante la Ilustre Universidad Simón Bolívar  
como requisito parcial para optar al título de

Ingeniero en Computación

**Sartenejas, Marzo de 2012**



UNIVERSIDAD SIMÓN BOLÍVAR  
DECANATO DE ESTUDIOS PROFESIONALES  
COORDINACIÓN DE INGENIERÍA DE COMPUTACIÓN

ACTA FINAL PROYECTO DE PASANTÍA

**DESARROLLO DE PROCESO DE SOLICITUD DE TARJETAS DE CRÉDITO  
MEDIANTE ARQUITECTURA SOA-BPM**

Presentado por:

HERNÁNDEZ LUIS

Este Proyecto de Grado ha sido aprobado por el siguiente jurado examinador:

Prof. Angela Di Serio

Prof. Edumilis Méndez

Lic. Edward Jaramillo

SARTENEJAS, 02/07/2012

## RESUMEN

El presente proyecto de pasantía se llevó a cabo con la finalidad de satisfacer la necesidad de una importante entidad bancaria a nivel nacional, la cual requería mejorar su proceso de solicitud de tarjetas de crédito, así como la automatización de ciertas actividades relacionadas que hasta ese momento se hacían de forma manual. Para llevar esto a cabo, IBM planteó la implementación y posterior implantación de un sistema *middleware* basado en la Arquitectura Orientada a Servicios y en la Gestión de Procesos de Negocio (SOA-BPM). Para desarrollar el sistema, se creó un proceso desde el cual se llamaba a una serie de servicios que se comunicaban con los sistemas legados para obtener la información requerida mediante un Bus de Servicios Empresariales que hacía posible esa comunicación. Como resultado, se logró automatizar puntos claves como el envío de la solicitud por Internet por parte del cliente lo que representaba una mejora ya que iniciaba su solicitud de forma remota, además fue una mejora para la empresa, ya que el personal que labora en esta entidad financiera ya no requería de almacenar las solicitudes por ellos mismos en sus servidores. Adicionalmente, mediante una herramienta asociada al servidor de procesos, el personal del banco disponía de una bandeja de entrada para cada una de las tareas del proceso, lo que genera un mayor orden a la hora de distribuir las actividades a lo largo de una solicitud, todo esto aunado a una interfaz amigable que reemplazó al viejo sistema basado en el lenguaje RPG, representando una ganancia en tiempo tanto para el cliente como para los usuarios y por lo tanto una mayor eficiencia a lo largo de todo el proceso de una solicitud de tarjetas de crédito.

**Palabras clave:** SOA, BPM, Tarjeta de Crédito, Servicios, Entidad Bancaria.

## DEDICATORIA

*A Dios,*

*Quien ha estado conmigo en cada momento de mi vida.*

*A mi abuelo Luis Jorge Sousa,*

*Una de las personas más maravillosas que he conocido.*

## **AGRADECIMIENTOS Y RECONOCIMIENTOS**

En primer lugar quiero agradecer a mi familia por ser ejemplo de unión y apoyo, en especial a mis padres Carmen y Luis y a mi abuela Teresa porque han sido incondicionales en todo momento y me han brindado una educación ejemplar, haciendo todo lo que esté en sus manos para que yo esté en donde estoy en este momento.

A mis amigos, por tenderme una mano cuando necesité ayuda y celebrar cuando obtuve algún logro, en especial a Katherina, principal fuente de inspiración y que ha estado a mi lado en los momentos en que más lo he necesitado.

A mi tutor académico, Edumilis Méndez por su completa disposición a responder mis dudas.

A mi tutor industrial Edward Jaramillo, así como a Maria Esther Barreiro por su comprensión y completa disposición en aquellos momentos en que necesité ayuda.

A mis compañeros Jenny, Lorena, Jonatán, John, Javier, Julio y Thomas por aportarme su gran experiencia y apoyo en cada momento durante mi estadía en IBM.

## ÍNDICE GENERAL

INDICE DE IMAGENES .....	IX
LISTA DE ABREVIATURAS .....	XI
INTRODUCCION .....	1
CAPÍTULO I .....	3
ENTORNO EMPRESARIAL.....	3
1.1 Antecedentes Históricos. ....	3
1.2 Participación en el mercado. ....	4
1.3 Estructura Corporativa. ....	4
CAPÍTULO II.....	6
ASPECTOS TEÓRICOS Y TECNOLÓGICOS DEL PROYECTO. ....	6
2.1 Servicio. ....	6
2.1.1 Interfaz.....	6
2.1.2 Servicios Web.....	6
2.2 Procesos de Negocio. ....	8
2.3 Arquitectura Orientada a Servicios (SOA): .....	9
2.4 Gestión de Procesos de Negocio.....	10
2.4.1 Lenguaje de Ejecución de Procesos de Negocio. ....	11
2.5 SOA-BPM.....	13
2.6 <i>Bus de Servicios Empresariales</i> .....	14
2.7 <i>Arquitectura de componentes de servicio (SCA)</i> .....	15
2.7.1 Componentes de servicio:.....	16
2.8 Modelado de Componentes de Negocio (CBM).....	17
2.9 Herramientas utilizadas en el proyecto: IBM WebSphere.....	18

2.9.1	Servidor de Aplicaciones de WebSphere (WAS).....	19
2.9.2	Servidor de Procesos de WebSphere (WPS).....	19
2.9.3	Desarrollador de Integraciones de WebSphere (WID).....	20
2.9.4	WebSphere Message Broker. ....	21
2.9.5	Colas de Mensaje de WebSphere (WebSphere MQ).....	21
2.10	Framework de la Industria de IBM (IFW). ....	22
CAPÍTULO III.....		23
MARCO METODOLÓGICO.....		23
3.1	Fase de Modelado de negocio y transformación. ....	23
3.2	Fase de Identificación. ....	24
3.3	Fase de Especificación.....	24
3.4	Fase de Realización e Implementación.....	25
4.CAPITULO IV: RESULTADOS OBTENIDOS EN BASE A SOMA.....		26
4.1	Fase 1: Modelado.....	26
4.1.1	Estudio de las herramientas involucradas en la solución. ....	26
4.1.2	Estudio de la arquitectura definida para el proyecto. ....	27
4.2	Fase 2: Identificación.....	29
4.2.1	Estudio de la funcionalidad existente en el sistema legado.....	29
4.2.2	Identificación de servicios candidatos.....	30
4.2.3	Elaboración del portafolio de servicios. ....	31
4.3	Fase 3: Especificación. ....	32
4.3.1	Elaboración de la Especificación de servicios.....	32
4.3.2	Creación de la matriz de datos de cada uno de los servicios.....	33
4.3.3	Creación del modelo de componentes.....	34

4.3.4	Definición del proceso a utilizar en el desarrollo. ....	35
4.4	Fase 4: Realización e Implementación. ....	41
4.4.1	Desarrollo del prototipo para la Prueba de Concepto. ....	41
4.4.2	Definición del protocolo de intercambio de mensajes con el sistema legado. ....	43
4.4.3	Desarrollo completo del proceso. ....	44
4.5	Fase 6: Despliegue. ....	47
4.6	Logros Adicionales: Elaboración de Portlets para un Módulo de Consulta de Requisiciones de Compra. ....	47
CONCLUSIONES Y RECOMENDACIONES .....		51
REFERENCIAS BIBLIOGRAFICAS. ....		53
Apéndice A: Tabla de Servicios Desarrollados. ....		55
Apéndice B: Descripción de Subsistemas .....		59
Apéndice C: Ejemplo de la información de un Servicio en el Portafolio de Servicios .....		60
Apéndice D: Modelo de Especificación de Servicios.....		71
Apéndice E: Modelo de Componentes .....		88



## INDICE DE IMAGENES

Figura 2.1. Representación de un proceso de negocio y sus elementos: Referencias, Interfaces y Enlaces.....	9
Figura 2.2. Interacción típica basada en SOA. ....	10
Figura 2.3. Implementación en BPEL. Recepción de un objeto, asignación de valor a dicho objeto, llamado de un servicio externo utilizando como entrada el objeto recibido y finalmente respuesta determinada.....	11
Figura 2.4. Interacción entre SOA y BPM. ....	14
Figura 2.5. ESB como mediador entre proveedor y consumidor de servicios. ....	15
Figura 2.6. Diagrama de Bloque de un componente de servicio.....	16
Figura 2.7. Mapa de componentes de Negocio CBM.....	18
Figura 3.1. Etapas SOMA involucradas en el proyecto. ....	23
Figura 4.1. Elementos involucrados en la arquitectura de la solución (SOA-BPM). (Elaboración Propia) .....	28
Figura 4.2. Diseño de la solución a alto nivel. (Elaboración Propia).....	29
Figura 4.3. Modelo de Matriz de datos para Consultar Cliente en el sistema legado X. ....	34
Figura 4.4. Subsistemas de la solución.....	35
Figura 4.5. Etapas del proceso de solicitud de TDC. ....	36
Figura 4.6. Diagrama de modelado de alto nivel del ingreso de la solicitud por parte del cliente. ....	37
Figura 4.7. Diagrama de modelado de alto nivel de las operaciones realizadas por el Especialista de negocios. ....	38
Figura 4.8. Diagrama de modelado de alto nivel de la etapa 3 de la solicitud de TDC. Analista I. ....	38
Figura 4.9. Diagrama de modelado de alto nivel de la etapa 4 de la solicitud de TDC. Analista de	

crédito. ....	39
Figura 4.10. Diagrama de modelado de alto nivel de la etapa 5 de la Solicitud de TDC. Nivel de autonomía. ....	40
Figura 4.11. Diagrama de modelado de alto nivel de la etapa 6 de la solicitud de TDC. Grabación y Distribución. ....	40
Figura 4.12. Etapas del desarrollo de la Prueba de Concepto. ....	42
Figura 4.13. Flujo de solicitud a nivel del <i>broker</i> . ....	43
Figura 4.14. Flujo de respuesta a nivel del <i>broker</i> . ....	44
Figura 4.15. Diagrama de ensamblaje del proceso a nivel del WPS. ....	45

## LISTA DE ABREVIATURAS.

XML	eXtended Markup Language
WID	WebSphere Integration Developer
WMB	WebSphere Message Broker
WSDL	Web Service Description Language
WAS	WebSphere Application Server
WPS	WebSphere Process Server
SOA	Service Oriented Architecture
SOMA	Service Oriented Modeling Architecture
ESB	Enterprise Service Bus
IBM	International Business Machines
BPM	Business Process Management
JSF	Java Server Faces
PCML	Process Call Markup Language
WS	Web Service.
BPEL	Business Process Execution Language
UDDI	Universal Description Discovery and Integration
URI	Uniform resource Identifier

CORBA	Common Object Request Broker Architecture
HTTP	HyperText Transfer Protocol
J2EE	Java 2 Enterprise Edition
IT	Information Tecnology
W3C	World Wide Web Consortium
XPath	XML Path Language
EIS	Executive Information System
OAM	Object Authority Manager
CBM	Component Business Modeling
UML	Unified Modeling Language
PoC	Proof of Concept (Prueba de Concepto)

## INTRODUCCION

La Arquitectura Orientada a Servicio en conjunto con la Gestión de Procesos de Negocio (SOA-BPM) es un paradigma que está en auge actualmente y son relativamente pocos los proyectos a nivel nacional que han utilizado esta forma de desarrollo, en gran parte por desconocimiento de la misma. International Business Machines (IBM), ha sido pionero en la promoción de SOA-BPM, aplicándolo tanto a nivel nacional como a nivel internacional, obteniendo hasta ahora buena retroalimentación por parte de los clientes que han aplicado a su empresa proyectos de esta naturaleza.

En base a esto, una importante entidad bancaria a nivel nacional, con más de 50 años de actividad ininterrumpida, y que se encuentra entre las 10 primeras instituciones del sector bancario venezolano, solicitó a IBM la creación de un sistema que ayude a mejorar el proceso de solicitud de tarjetas de crédito, el cual, se realiza en gran medida de forma manual, pudiendo ser automatizado, y en donde además, se le otorgue al personal que labora en el banco, un medio para gestionar las tareas humanas que éste tenga.

Para satisfacer esta necesidad, IBM ofreció a dicha entidad financiera, un sistema *Middleware* basado en la Arquitectura Orientada a Servicios, que ofreciera al personal que allí labora una interfaz amigable, y en donde además se automatizaran aquellos procesos que aún no lo están, centralizando en una sola herramienta la realización de todas las tareas correspondientes a la consulta de información en los sistemas legados. A continuación se presentan los objetivos que tuvo este proyecto de pasantía.

El logro de este proyecto de pasantía aportó un sistema que automatizó una serie de actividades que se realizaban de forma manual como:

- La generación de la solicitud por vía electrónica.
- Acceder a una sola aplicación para el proceso de análisis.
- Automatización de la gestión de los procesos.
- Mejora en el proceso de clasificación de tarjetas de crédito.

A continuación, se presentan los objetivos de este proyecto de pasantía.

### **Objetivo General.**

Enmarcado dentro del modelado de procesos que permitieran desarrollar el sistema propuesto/ofrecido al cliente, este proyecto de pasantía se planteó como objetivo general Desarrollar las funcionalidades asociadas al proceso de solicitud de tarjetas de crédito, y cuyos objetivos específicos son:

- Analizar la funcionalidad existente.
- Estudiar la arquitectura del sistema.
- Elaborar el modelo de negocio.
- Elaborar la matriz de datos para cada uno de los servicios.
- Implementar los servicios correspondientes a la Prueba de Concepto.
- Implementar el proceso y los servicios correspondientes al proceso de solicitud de Tarjetas de Crédito.
- Generar la documentación correspondiente.

A continuación, en el Capítulo I se dará una descripción de IBM como empresa. Luego, en el Capítulo II se explicará brevemente conceptos teóricos de suma importancia para entender un poco más el paradigma de desarrollo SOA-BPM, para luego, en el Capítulo III describir cada una de las fases por las que pasó el proyecto así como la metodología utilizada. Finalmente, en el Capítulo IV se presentarán los resultados de este trabajo y las recomendaciones pertinentes, así como los logros adicionales obtenidos a lo largo de la pasantía.

## **CAPÍTULO I**

### **ENTORNO EMPRESARIAL**

International Business Machines (IBM, por sus siglas en inglés) es una empresa multinacional con presencia principal en prácticamente todos los segmentos relacionados con las tecnologías de la información, siendo una gran parte de sus ingresos provenientes de las ramas de consultoría y servicios.

Este capítulo está dedicado a la descripción de la compañía IBM, desde una breve reseña histórica y estructura corporativa, hasta la importancia dentro del mercado global y la unidad en la que se llevó a cabo el proyecto de pasantía.

#### **1.1 Antecedentes Históricos.**

La compañía IBM fue creada el 15 de Junio de 1911 en Binghamton (Estados Unidos) como una fusión gestionada por Charles R. Flint de su empresa Tabulating Machine Company, con Computing Scale Corporation e International Time Recording Company. La empresa formada de la fusión fue llamada Computing Tabulating Recording Corporation (CTR) y estuvo bajo el mando de Thomas J. Watson, pero el 14 de febrero de 1924 CTR cambió su nombre a International Business Machines Corporation (IBM), se dice que debido a una especie de broma, al ser líder del sector la compañía de máquinas registradoras "National".

Las empresas originarias de CTR fabricaban una amplia gama de productos, desde sistemas para el control de empleados hasta equipos automatizados para el corte de carne. Adicionalmente, fabricaban equipos para la gestión de tarjetas perforadas, que serían un elemento clave de los futuros computadores. Con el tiempo, CTR se centraría en estos equipos y dejaría a un lado la fabricación de sus otros productos.

A partir de la década de los 40, IBM comenzó a investigar en el campo de la informática.

En 1944 terminó de construir el computador Automatic Sequence Controlled Calculator (Calculadora Controlada por Secuencia Automática), también conocido como Mark I. El Mark I fue la primera máquina capaz de ejecutar cálculos complejos automáticamente basada en interruptores electromecánicos.

IBM llega a Venezuela el 18 de Enero de 1938 cuando CA Watson de Máquinas Comerciales abrió sus puertas en Caracas. Desde entonces, ha conquistado el mercado venezolano de diversas formas: desde la instalación en la década de los cincuenta de sistemas que transformaron las plataformas tecnológicas de industrias como la financiera y la petrolera, hasta la producción de computadores personales, la implementación de supercomputadores y la inauguración de su Centro de Innovación de Negocios. Actualmente, la torre IBM Venezuela se encuentra en la ciudad de Caracas, en la Avenida Ernesto Blohm en Chuao <sup>[1]</sup>.

## **1.2 Participación en el mercado.**

Actualmente, alrededor de 400,000 IBMistas alrededor del mundo crean e integran hardware, software y servicios para ayudar a empresas, instituciones y personas con visión de futuro a ser exitosos en un mundo inteligente.

IBM opera en más de 170 países y disfruta de una creciente distribución de la renta, agrupando los mercados por las características de crecimiento común y no por la ubicación geográfica.

Adicionalmente, IBM mantiene el más grande cuerpo de investigación a nivel mundial. En 2008, por decimosexto año consecutivo, IBM fue la empresa que recibió más patentes en los Estados Unidos. En los últimos 6 años IBM realizó una agresiva inversión de 35 billones de dólares en investigación y desarrollo con más de 100 adquisiciones de compañías de hardware, servicios y software <sup>[1]</sup>.

## **1.3 Estructura Corporativa.**

IBM está principalmente estructurado por 6 departamentos:

- **Servicios Globales de Tecnología** (GTS, por sus siglas en inglés): constituye el área



que IBM coloca a su disposición para proporcionar los más completos y confiables servicios de tecnología del mercado, destinados a satisfacer las necesidades tecnológicas de sus clientes, cumpliendo con los estándares y mejores prácticas de la industria.

- **Servicios Globales de Negocio** (conocido como GBS): IBM GBS es la organización de servicios de consultoría más grande del mundo, que brinda a sus clientes su experiencia en procesos de negocios y en la industria con un profundo entendimiento de las soluciones tecnológicas apuntadas a problemas específicos, junto a la habilidad para diseñar, construir e implementar dichas soluciones para otorgar valor al negocio.
- **Sistemas y Tecnología**: Provee una avanzada computación, almacenamiento y tecnología de semiconductores y productos.
- **Software Group**: Provee *middleware* que permite al cliente integrar sistema, procesos y aplicaciones, así como sistemas operativos.
- **Finanzas Globales**: Crea soluciones financieras que pueden ser modificadas para solucionar necesidades de TI específicas, desde tasa de competitividad, hasta planes de pago flexibles o préstamos.
- **Ventas, investigación y desarrollo**: Provee un ancho espectro de apoyo al cliente en las soluciones y valor adicional <sup>[1]</sup>.

Este proyecto se realizó para el departamento de GBS, que es el encargado de desarrollar servicios de consultoría para los clientes externos. El pasante estuvo ocupando el rol de Consultor.

## CAPÍTULO II

### ASPECTOS TEÓRICOS Y TECNOLÓGICOS DEL PROYECTO.

En este capítulo se introducirán los conceptos relacionados tanto con la Arquitectura Orientada a Servicios (SOA por sus siglas en inglés), como con la Gestión de Procesos de Negocio (BPM, por sus siglas en inglés), así como las definiciones asociadas a dicho paradigma de programación. Adicionalmente, desde el punto de vista tecnológico se dará una descripción de las herramientas involucradas en el proceso.

#### 2.1 Servicio.

Un servicio es una tarea de negocio que se puede repetir, por ejemplo, chequear el crédito de un cliente. Es la entidad lógica que representa el contrato definido por una o más interfaces <sup>[2]</sup>..

##### 2.1.1 Interfaz.

Define un conjunto de firmas públicas de métodos, lógicamente agrupadas pero sin implementación. Una interfaz define un contrato entre el solicitante y el proveedor del servicio. Una implementación de una interfaz debe proveer todos los métodos <sup>[2]</sup>..

##### 2.1.2 Servicios Web.

Un Servicio Web es una aplicación de Software identificada por un identificador único (URI en sus siglas en inglés por Uniform Resource Identifier), cuyas interfaces y asociaciones son capaces de ser definidas, descritas y descubiertas como artefactos XML. Un servicio web soporta interacciones directas con otros agentes de software usando mensajes XML, interactuando a través de protocolos basados en Internet <sup>[2]</sup>..

Entre las características claves de los servicios web están <sup>[2]</sup>.

- **Los servicios son auto-contenidos:** Del lado del cliente, no hace falta algún software adicional. Un lenguaje de programación que soporte XML y HTTP, por

ejemplo es suficiente para trabajar con Servicios Web. Del lado del servidor, un servidor web y un motor de *servlets* es requerido. Es posible que los servicios web activen una aplicación existente sin escribir una línea de código.

- **Los servicios web son auto-descriptores:** Ni el cliente ni el servidor saben o precisan alguna información adicional que no sea el formato y el contenido de los mensajes de solicitud y respuestas. La definición del formato viaja con el mensaje, no son requeridos repositorios eternos de metadata ni herramientas de generación de código.

- **Los servicios web son modulares:** Los servicios web son una tecnología para construir y otorgar acceso a funciones de negocio a través de la Web; J2EE, CORBA y otros estándares son tecnologías para implementar servicios web.

- **Los servicios web pueden ser publicados, localizados e invocados a través de la Web:** Los estándares requeridos para hacer esto son:

- **Protocolo de Acceso Simple a Objetos (SOAP, por sus siglas en inglés):** Es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

- **Lenguaje de Descripción de Servicios Web (WSDL, por sus siglas en inglés):** es un formato XML para describir servicios web como un conjunto de destinos operando en mensajes que contienen información orientada a documentos u orientada a procesos.

- **Descripción Universal, Descubrimiento e Integración (UDDI, por sus siglas en inglés):** Es un mecanismo de registros que puede ser usado para buscar descripciones de servicios web.

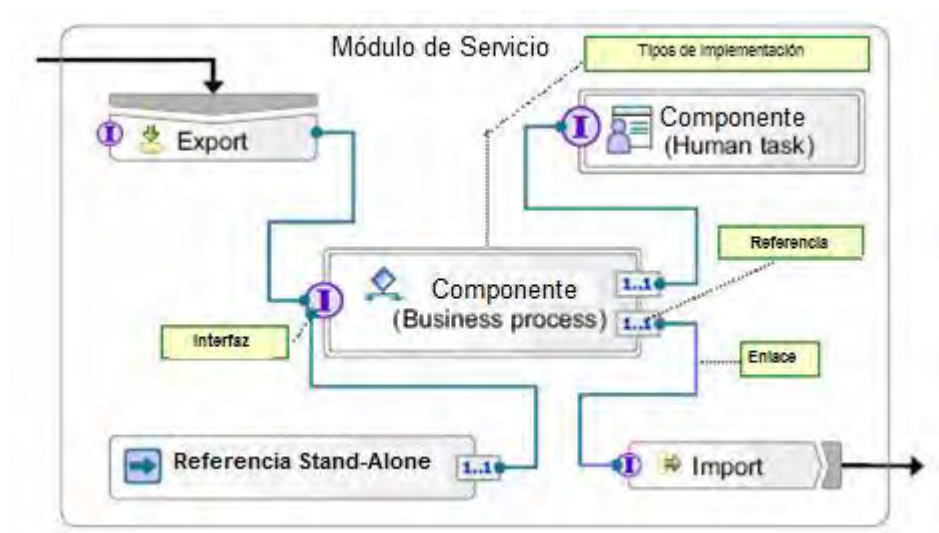
- **Los servicios web son independientes del lenguaje de programación e interoperables:** La interacción entre un proveedor de servicios y un solicitante es diseñada para ser totalmente independiente de la plataforma y el lenguaje. Esta interacción requiere un documento WSDL para definir la interfaz y describir el servicio, junto con el protocolo de red (generalmente HTTP). Debido a que ni el proveedor de servicio ni el solicitante del servicio tienen idea de que plataformas o lenguajes está usando el otro, la interoperabilidad es un hecho.

- **Los servicios web son abiertos a herencia y basados en estándares:** XML y HTTP son fundamentos técnicos para los Servicios web. Una gran parte de las tecnologías de Servicios Web han sido construidas usando proyectos de código abierto. Por tanto, la independencia del proveedor y la interoperabilidad son metas realísticas.
- **Los Servicios Web son dinámicos:** Negocios electrónicos dinámicos pueden ser una realidad usando Servicios Web porque con UDDI y WSDL, la descripción y descubrimiento de los Servicios Web puede ser automatizada.
- **Los Servicios Web pueden ser compuestos:** Servicios Web simples pueden ser agregados a algún otro más complejo, ya sea usando técnicas de *workflow* o llamando Servicios Web de alguna capa inferior desde alguna implementación de un Servicio Web.<sup>[2]</sup>

## 2.2 Procesos de Negocio.

Un proceso de negocio es una colección de actividades o tareas estructuradas y relacionadas que generan un producto o servicio específico para un cliente o grupo de clientes particular. Los procesos de negocio pueden ser visualizados como una secuencia de actividades.

En la figura 2.1 se muestra una representación de un proceso de negocio. El componente “Proceso de Negocio” se representa como pieza central del producto o servicio, a la izquierda, se encuentran las interfaces asociadas donde generalmente se expone el servicio y a la derecha, se observan las referencias, es decir, aquellos servicios que son llamados para ser utilizados en la implementación del Proceso de Negocio.

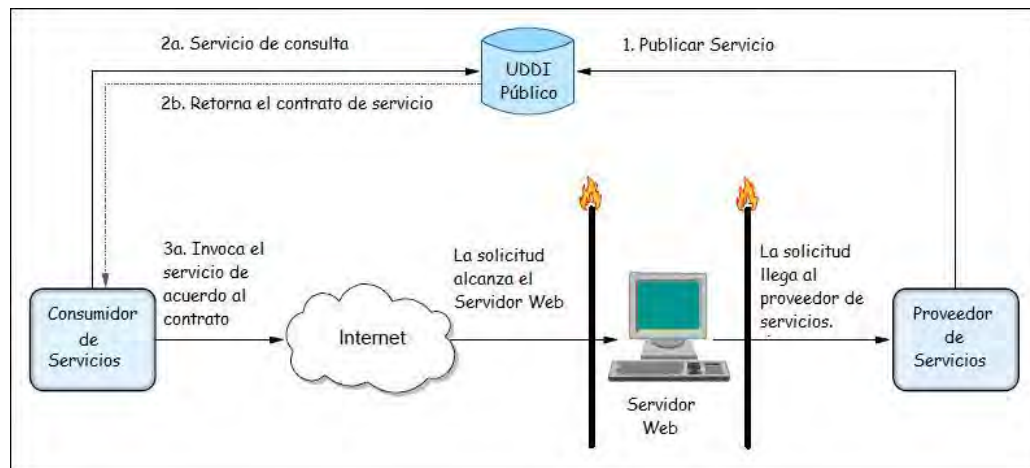


**Figura 2.1.** Representación de un proceso de negocio y sus elementos: Referencias, Interfaces y Enlaces [2].

### 2.3 Arquitectura Orientada a Servicios (SOA):

En rasgos generales, SOA es un estilo arquitectural de la tecnología de la información que tiene como finalidad integrar el negocio del cliente como servicios adjuntos. SOA facilita dicha integración en un proceso de negocio como ensamblar bloques en una estructura. También, permite a los servicios comunicarse con otros usando el ambiente de comunicación abierta que ellos crean. Los servicios interactúan para que IT soporte procesos de negocio interrelacionados. No importa qué tan diferente sean sus sistemas y tecnologías, SOA hace posible compartir información, no sólo a través de una empresa, sino con clientes, proveedores y socios [2].

La figura 2.2 muestra la interacción típica de un proceso bajo la Arquitectura Orientada a Servicios. En primer lugar, el proveedor de servicios publica un servicio en el UDDI público, luego el consumidor de servicios hace una consulta al UDDI y recibe el contrato de servicio. Al recibirlo, hace la invocación del servicio a través de internet, y a través de un servidor web, la solicitud llega al proveedor de servicios [2].



**Figura 2.2.** Interacción típica basada en SOA [2].

Basado en SOA, un consumidor de servicio no tiene que preocuparse por algún servicio con el que se esté comunicando, ya que la infraestructura o “bus” de servicio hará una elección apropiada por el consumidor.

La infraestructura esconde de un solicitante la mayor cantidad de tecnicismos posibles. Particularmente, especificidades técnicas de diferentes tecnologías de implementación como J2EE o .NET no afectará a los usuarios SOA. Por último, permite también reconocer y sustituir una implementación de servicio mejor -si existe- y con mejores características de calidad de servicio [2].

## 2.4 Gestión de Procesos de Negocio.

Se llama Gestión de Procesos de Negocio (BPM) a la metodología corporativa cuyo objetivo es mejorar la eficiencia a través de la gestión de los procesos de negocio que se deben modelar, organizar, documentar y optimizar de forma continua. Como su nombre

sugiere, BPM se enfoca en la administración de los procesos dentro de una organización. Es un conjunto de recursos y actividades interrelacionadas que transforman elementos de entrada en elementos de salida.

A través del modelado de las actividades y procesos puede lograrse un mejor entendimiento del sistema y muchas veces esto presenta una oportunidad para mejorarlos [3].

#### 2.4.1 Lenguaje de Ejecución de Procesos de Negocio.

El Lenguaje de Ejecución de Procesos de Negocio (BPEL) es un lenguaje basado en XML que modela un proceso de negocio como la composición de un conjunto de servicios web elementales. El llamado motor BPEL es un componente de software dedicado que permite ejecutar definiciones de procesos BPEL. A cada proceso también puede accederse como un servicio web del motor BPEL. La especificación depende de los estándares W3C para las descripciones de los servicios web, esquemas XML para la definición de las estructuras y XPath para la recuperación de elementos XML. En la Figura 2.3 se muestra la implementación de un proceso de negocio utilizando el BPEL (Business Process Execution Language) [3].



**Figura 2.3.** Implementación en BPEL. Recepción de un objeto, asignación de valor a dicho objeto, llamado de un servicio externo utilizando como entrada el objeto recibido y finalmente respuesta determinada. [4]

### 2.4.1.1 Componentes del BPEL

Los principales componentes del BPEL son [3]:

- **Servicios Asociados (*Partner Links*):** Provee un canal de comunicación a un servicio web remoto el cual es usado en el proceso BPEL. Un tipo respectivo de servicio asociado puede ser definido primero para especificar el puerto WSDL requerido y provisto.
- **Variables:** Son usadas para almacenar tanto datos de mensajes de las interacciones de servicios web, como datos de control del proceso. Una variable puede ser declarada en el encabezado de un proceso BPEL referenciando un WSDL o un tipo de datos de esquemas XML.
- **Correlación:** Como BPEL soporta procesos de negocio de larga ejecución, pueden haber varias instancias de procesos esperando por mensajes de servicios web en un cierto momento. Un conjunto de correlación especifica las llamadas propiedades, por ejemplo, sentencias XPath, para recuperar partes de mensajes que son únicas para una instancia de proceso específica. De acuerdo con un cierto valor de propiedad, un mensaje es enviado a la instancia del proceso correspondiente.
- **Actividades Básicas:** Son los pasos básicos del BPEL. Existen actividades para enviar y recibir mensajes desde los servicios web (*Receive*, *Invoke*, *Reply*), para cambiar el contenido de variables (*Assign*), para esperar un cierto período o a un cierto punto en el tiempo (*wait*), o para terminar el proceso (*exit*, llamado formalmente *terminate*).

El flujo de control de las actividades básicas puede ser definido en dos estilos diferentes: orientado a bloques o basados en grafos, ambos estilos pueden ser mezclados. El flujo de control orientado a bloques es definido mediante actividades estructuradas mientras que, el flujo de control basado en grafos, usa las asociaciones (sincronización entre dos actividades) pero sin introducir ciclos.

- **Actividades Estructuradas:** BPEL ofrece actividades para especificar una



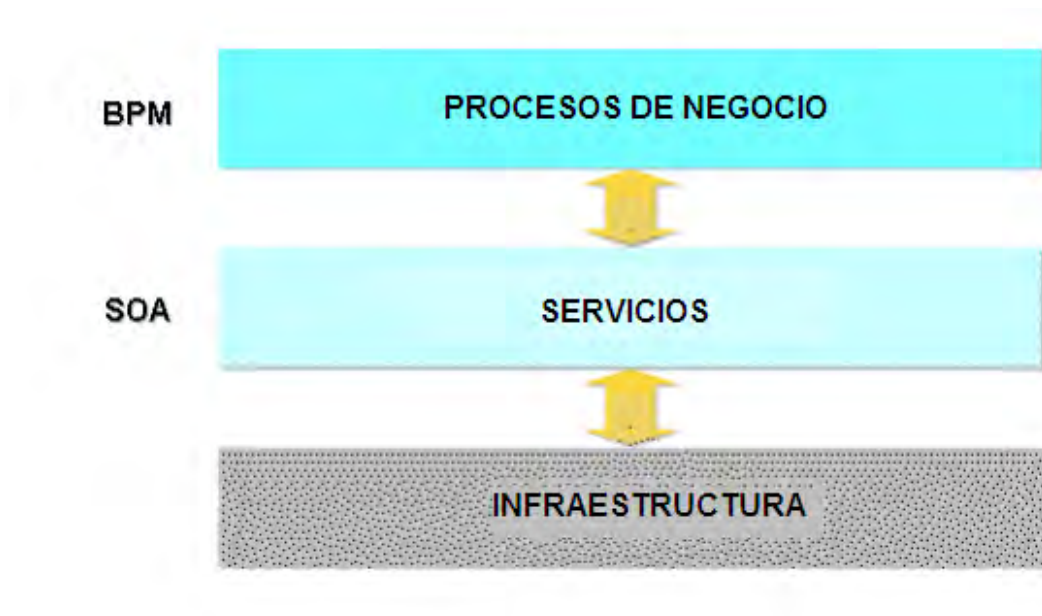
ejecución paralela (*flow*), saltos condicionales basados en data (*if-else*) o en la recepción de un mensaje (*pick*), ejecución secuencial (*sequence*) y de diferentes ciclos (*while*, *repeatUntil*, *forEach*). A su vez, las actividades estructuradas pueden ser solapadas.

Los alcances son actividades estructuradas especiales que marcan el alcance de variables locales y manejadores. **Manejadores:** BPEL provee manejadores para solucionar situaciones inesperadas o excepcionales. Los manejadores de eventos esperan por eventos de mensaje o de tiempo. Pueden ser usados para especificar una ruptura en el nivel del proceso. Los manejadores de errores captan errores internos del proceso BPEL. Si el error no puede ser solucionado, el manejador de compensación puede ser disparado para deshacer los efectos de actividades ya terminadas. Finalmente, el manejador de terminación ofrece un mecanismo para forzar a los procesos a terminar, por ejemplo, debido a faltas externas<sup>[3]</sup>.

## 2.5 SOA-BPM

El constructo principal en SOA y BPM son los servicios y procesos respectivamente, sin embargo, esta distinción pierde relevancia con el hecho de que los procesos de por sí pueden ser publicados como servicios. Desde una perspectiva de alto nivel, SOA maneja servicios y BPM procesos de negocio. Los procesos de negocio son hechos a partir de servicios, lo que puede incluir tareas humanas y otros procesos de negocio. Los servicios, en cambio son hechos de otros servicios de bajo nivel que están unidos a una infraestructura determinada. [5].

En la figura 2.4 se puede ver la relación entre SOA y BPM. En el nivel más alto se encuentran los procesos de negocio, desde ahí una serie de servicios son llamados, y éstos, al ser llamados se comunican con la infraestructura para obtener la información necesaria y devolverla al proceso de negocio para que continúe su flujo.



**Figura 2.4.** Interacción entre SOA y BPM. [5].

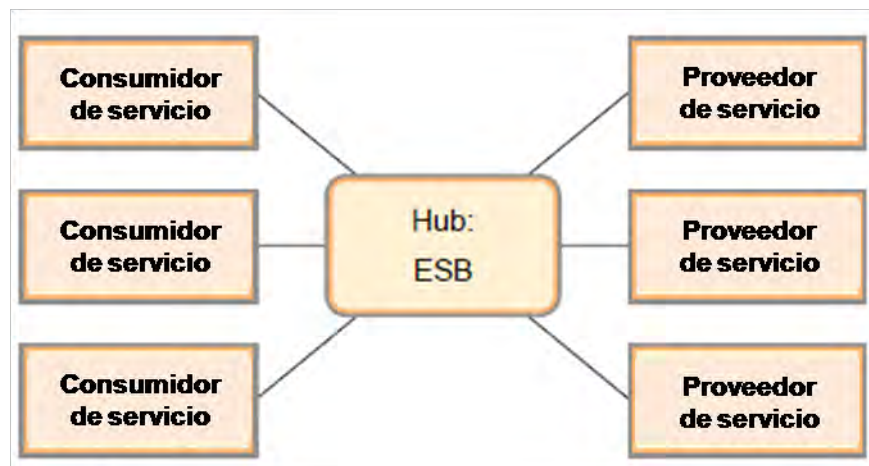
## 2.6 *Bus de Servicios Empresariales*

El Bus de Servicios Empresariales (ESB, por sus siglas en inglés) está emergiendo como un componente con infraestructura de *middleware* que soporta la implementación de SOA en una empresa. ESB soporta los conceptos de la implementación de SOA al [6] :

- Separar la vista del consumidor de un servicio de su implementación en la realidad.
- Separar aspectos técnicos de interacciones de servicio.
- Integrar y manejar servicios en la empresa.

Estos conceptos son logrados reemplazando conexiones directas entre los proveedores y los consumidores de servicio [6].o

En la Figura 2.5 se puede ver como el ESB funge de mediador entre los proveedores y consumidores de servicio.



**Figura 2.5.** ESB como mediador entre proveedor y consumidor de servicios. [6].

El ESB puede ser usado para emplear una de las siguientes funciones de *middleware* [6] :

- Adaptar solicitudes de servicio de un protocolo y dirección a otro.
- Transformar formatos de datos.
- Soportar una variedad de modelos de seguridad y transaccionales entre los consumidores y proveedores y reconocer que ambos pueden soportar o requerir diferentes modelos.
- Agregar o desagregar solicitudes y respuestas de servicios.
- Soportar protocolos de comunicación entre múltiples plataformas con la calidad de servicio apropiada.
- Proveer capacidades de mensajería como la correlación de mensaje y el “publicar/suscribir” para soportar diferentes modelos de mensajes, como eventos o solicitudes/respuestas asíncronas.

## **2.7 Arquitectura de componentes de servicio (SCA)**

La Arquitectura de Componentes de Servicios (SCA, por sus siglas en ingles) es un

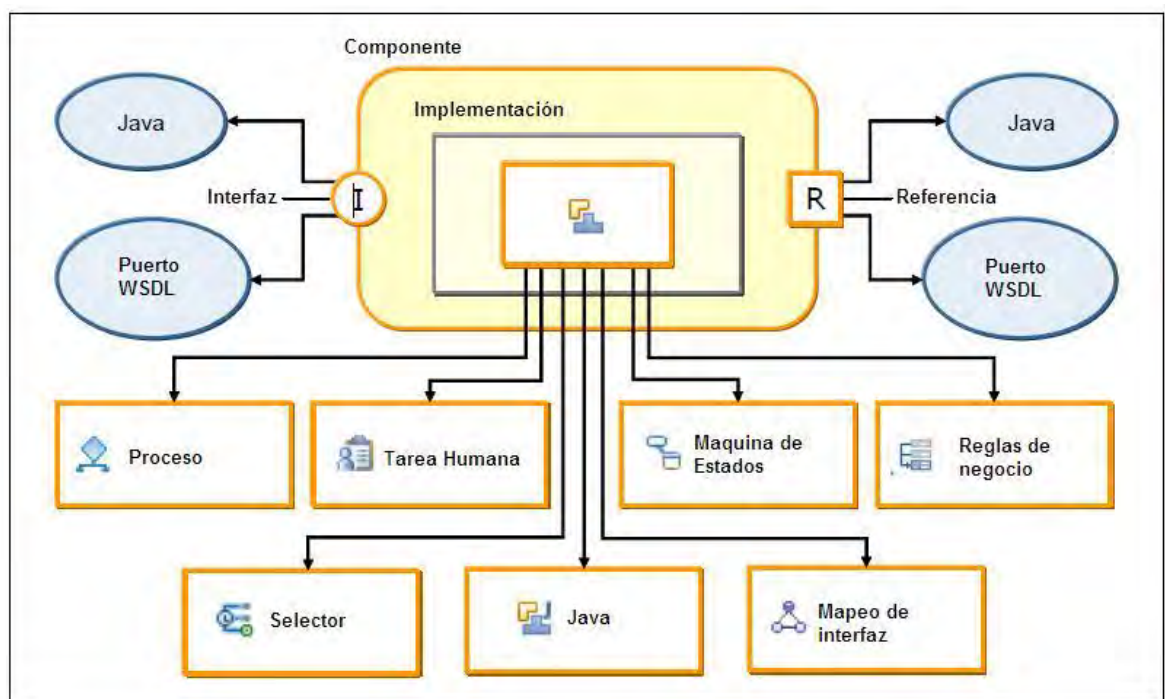
conjunto de especificaciones que describen un modelo para construir aplicaciones y sistemas usando SOA. La SCA establece un código de organización de las aplicaciones de negocios basado en componentes que implementan la lógica. Esas aplicaciones de negocio ofrecen sus capacidades como servicios a través de interfaces y consumen servicios que son ofrecidos por otros a través de referencias.

SCA separa la lógica de negocio de la implementación. La implementación de procesos de negocio está contenida en los componentes de servicio, los cuales pueden ser ensamblados gráficamente en WebSphere Integration Developer (WID) [6].

### 2.7.1 Componentes de servicio:

Un componente de servicio consta de: una implementación, una o más interfaces que definen entradas, salidas y errores, y cero o más referencias. Una referencia identifica a la interfaz perteneciente a otro servicio o componente que pueda ser requerida o consumida por otro componente de servicio [6].

En la Figura 2.6, se diagraman los elementos que constituyen un componente de servicio.



**Figura 2.6.** Diagrama de Bloque de un componente de servicio. [6].

## 2.8 Modelado de Componentes de Negocio (CBM).

El Modelado de Componentes de Negocio (CBM, por sus siglas en ingles) es una técnica para analizar una empresa haciendo una partición en componentes de negocio independientes y no solapados para identificar oportunidades para innovación y/o mejora <sup>[7]</sup> .

Un componente de negocio es<sup>[7]</sup> :

- Parte de una empresa que tiene el potencial de operar independientemente, en casos extremos como una compañía separada o como parte de otra compañía.
- Una vista lógica de la parte de una empresa que incluye los recursos, personal, tecnología, y los conocimientos necesarios para entregar algún valor.
- Un bloque de construcción o habilidad que crea valor en la organización.
- Un aspecto de la empresa que agrega valor ofreciendo productos o servicios a otros componentes de negocios, algunos de los cuáles están fuera de la empresa.

En la vista de CBM, una empresa es simplemente una colección de componentes de negocios que están conectados en red. CBM crea un Mapa de Componentes de Negocio (Figura 2.7) de los bloques de construcción esenciales de la organización en una estructura tabular que puede fácilmente encajar en una sola página.

	Desarrollo del Negocio	Relación con el cliente	Tecnología de Información	Gerencia Financiera	Operaciones de Negocio
Dirigir	Estrategia de Mercadeo	Estrategia del proveedor	Estrategia TI	Estrategia de Inversión	Estrategia de Organización y Operación
	Planificación de Adquisiciones	Estrategias del Empleador	Solicitudes		Estrategias de Regulación
Controlar	Gestión del prog. de Mercadeo y ventas	Gerencia de Relaciones con el Proveedor	Gerencia de Operaciones	Auditorías internas	Gestión de reclamos
	Gestión de Adquisiciones	Servicios de Proveedor	Gestión del portafolio de aplicaciones	Conteo Financiero	Gerencia Legal
Ejecutar	Actividades de adquisición	Servicios de proveeduría	Mantenimiento /desarrollo de aplicaciones	Funciones de auditoría interna	Procesamiento de reclamos
	Entrega del programa de Mercadeo	Servicios de Empleo	Operaciones IT	Procesamiento de facturación y pagos	Servicios legales

Figura 2.7. Mapa de componentes de Negocio CBM<sup>[8]</sup>.

## 2.9 Herramientas utilizadas en el proyecto: IBM WebSphere

Para la realización de este proyecto, se utilizó el paquete de herramientas WebSphere de IBM, el cual permite construir aplicaciones y soluciones empresariales y combinarlas con funciones innovadoras. WebSphere incluye y soporta un rango de productos que ayudan al programador a desarrollar y gestionar las aplicaciones de negocios. Estos productos facilitan la construcción, desarrollo, y administración de sitios web dinámicos y otras soluciones más completas, productiva y efectivamente.

Entre las herramientas WebSphere utilizadas están:

- Servidor de Aplicaciones de WebSphere (WAS).
- Servidor de Procesos de WebSphere (WPS).
- IBM WebSphere Message Broker.
- Desarrollador de Integraciones de WebSphere (WID)
- Cola de Mensajes de WebSphere (WebSphere MQ)

- Modelador de Negocios de WebSphere (WBM)

A continuación se abordarán cada una de estas herramientas y se indicará su intervención, aporte o valor para la construcción de la solución propuesta.

### **2.9.1 Servidor de Aplicaciones de WebSphere (WAS).**

El Servidor de Aplicaciones de WebSphere es un ambiente de ejecución de IBM para aplicaciones basadas en Java que provee la infraestructura para ejecutar aplicaciones que manejan el negocio. Éste separa la infraestructura del hardware, Sistema Operativo y la red. Un servidor de aplicaciones sirve como una plataforma para desarrollar y construir los Servicios Web y EJBs (Enterprise Java Beans). Éste también sirve como un motor de transacciones y de mensajería. Un servidor de aplicaciones también entrega la lógica de negocio a los usuarios en varios dispositivos cliente<sup>[9]</sup>.

### **2.9.2 Servidor de Procesos de WebSphere (WPS).**

IBM WPS es el servidor de integración de procesos de empresa que se ha desarrollado a partir de conceptos de integración comprobados, tecnologías de servidor de aplicaciones y los últimos estándares abiertos.

El Servidor de Procesos de IBM, que da soporte a una arquitectura orientada a servicios (SOA), es la plataforma ideal para las aplicaciones de empresa que requieren la integración empresarial mediante distintas tecnologías. Utilizando el conjunto de herramientas del Desarrollador de Integración de WebSphere (WID, por sus siglas en inglés) pueden crearse soluciones de integración empresarial utilizando mecanismos de integración simplificados como, por ejemplo, el modelo de programación de la Arquitectura de Componentes de Servicio (SCA, por sus siglas en inglés).

Con el Servidor de Procesos de WebSphere (WPS, por sus siglas en inglés), las aplicaciones de integración empresarial pueden definir lógicas y procesos de empresa basados

en BPEL, tareas de usuarios y normas empresariales<sup>[10]</sup> .

### **2.9.3      Desarrollador de Integraciones de WebSphere (WID)**

El Desarrollador de Intergraciones WebSphere (WID por sus siglas en ingles), ha sido diseñado como un entorno de desarrollo de integración completo para construir aplicaciones integradas. Para simplificar y acelerar el desarrollo de estas, este entorno proporciona una capa de abstracción que separa los componentes presentados visualmente con los que trabaja desde la implementación subyacente.

Estas herramientas presentan aplicaciones, incluyendo aquellas que existen remotamente en sistemas EIS y procesos de negocio como componentes. Los componentes se crean y se ensamblan en otras aplicaciones integradas (es decir, creadas a partir de un conjunto de componentes) mediante editores visuales. Los editores visuales presentan una capa de abstracción entre los componentes y sus implementaciones. Un desarrollador que utilice estas herramientas puede crear una aplicación integrada sin el conocimiento detallado de la implementación subyacente de cada componente.

Para construir una aplicación integrada a través de estas herramientas existen dos métodos de diseño: descendente o ascendente. La diferencia entre los dos radica en que, en el descendente, no existe en principio la implementación para uno o varios componentes, ésta se añade posteriormente. Por el contrario, en el diseño ascendente, los componentes ya están implementados y el desarrollador los ensambla arrastrándolos y soltándolos en un editor visual, creando después un flujo lógico entre ellos uniéndolos con líneas. Tras la construcción de la aplicación, un entorno de depuración y prueba ejecuta una prueba completa antes de desplegar las aplicaciones en un servidor de producción; establecer puntos de supervisión permitirá ver cómo se utiliza una aplicación en tiempo real para ajustarla para un rendimiento óptimo.

Las herramientas de WID están basadas en una arquitectura orientada a servicios. Los componentes son servicios y, una aplicación integrada que incluye muchos componentes, es un servicio. Los servicios creados se ajustan a los estándares principales de la industria. Los



procesos de negocio, que también se convierten en componentes, se crean de forma parecida con herramientas visuales de fácil utilización que se ajustan al lenguaje estándar de la industria BPEL <sup>[11]</sup>.

#### **2.9.4 WebSphere Message Broker.**

El WebSphere Message Broker (WMB, por sus siglas en inglés) es un poderoso mediador de información, es decir, establece la comunicación entre el WebSphere Process Server y el sistema legado, y permite, tanto a la data de negocio como a la información, en la forma de mensajes, seguir un flujo entre aplicaciones disparadas y a través de múltiples plataformas de hardware y software. Las reglas de negocio pueden ser aplicadas a la data que está moviéndose a través del Broker para enrutar, almacenar, solicitar y transformar la información.

Las principales habilidades de WMB son enrutamiento, transformación y enriquecimiento del mensaje, así como la publicación y suscripción. Juntas esas habilidades hacen de WMB una herramienta poderosa para integración de negocios.

WebSphere Message Broker está comprendido por: un ambiente de desarrollo, para la creación de conjuntos y flujos de mensajes, y algún otro recurso de aplicaciones de flujo de mensajes, y un ambiente de ejecución, que contiene los componentes para correr esas aplicaciones de flujos de mensajes que son creadas en el ambiente de desarrollo. <sup>[12]</sup>

#### **2.9.5 Colas de Mensaje de WebSphere (WebSphere MQ).**

IBM WebSphere MQ es una plataforma establecida y confiable de *middleware* de encolamiento de mensajes. Una infraestructura de encolamiento de mensajes construida con la tecnología WebSphere MQ puede proveer un transporte de mensajes confiable, seguro, escalable y mantenible para mensajes con aseguramiento de entrega.

WebSphere MQ provee características para garantizar la seguridad de acceso,

autenticación de identidad y la seguridad e integridad de la comunicación. El manejador de autoridad de objetos (OAM) es el servicio de autorización por defecto para manejo de comandos y objetos. Todas las acciones realizadas por una aplicación conectada a un manejador de colas, son autenticadas por el OAM<sup>[13]</sup>.

## **2.10 Framework de la Industria de IBM (IFW).**

El IFW es una colección de modelos interrelacionados que se dirige a diferentes aspectos del análisis y el diseño de soluciones bancarias orientadas a servicios. Los modelos de industria soportan la identificación, análisis y diseño de soluciones basadas en requerimientos que se relacionan con el ambiente del proyecto. Los retos tecnológicos de la integración empresarial varían según la infraestructura seleccionada. Los retos de capturar y expresar requerimientos de negocios y de derivar una arquitectura orientada a servicios de esos requerimientos en cambio no varían. La finalidad de los modelos es servir de soporte en el análisis de los requerimientos y en la etapa de diseño para que no tienda hacia una arquitectura en particular. Adicionalmente, el resultado de estos artefactos puede ser transformado en dominios tecnológicos como BPEL o J2EE.

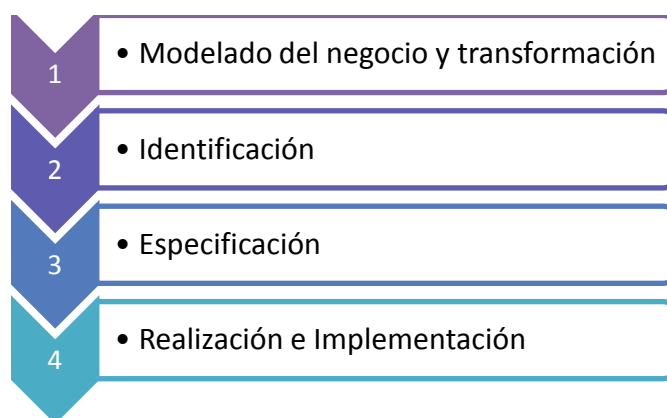
## CAPÍTULO III

### MARCO METODOLÓGICO

Para alcanzar el éxito en un proyecto de SOA-BPM, es fundamental seguir una metodología que garantice el correcto desenvolvimiento del proyecto en todas sus fases. Para el desarrollo de este proyecto de pasantía se utilizó la metodología de Arquitectura de Modelado Orientada a Servicios (SOMA ®).

SOMA® es un método de desarrollo e integración (Ciclo de vida del proceso) basada en principios de ingeniería de software para producir soluciones mediante la Arquitectura Orientada a Servicio (SOA). Éste método incluye orientación prescriptiva aplicable durante las fases e iteraciones requeridas en el ciclo de vida de un proceso de desarrollo de software.

A continuación, se explica el trabajo llevado a cabo durante cada una de las etapas de SOMA esquematizadas en la Figura 3.1. <sup>[14]</sup>



**Figura 3.1.** Etapas SOMA involucradas en el proyecto. (Basado en [15])

#### 3.1 Fase de Modelado de negocio y transformación.

Durante esta fase se realizó un análisis de la funcionalidad del sistema de solicitud de tarjetas de crédito que estaba siendo utilizado en la entidad financiera, así como un estudio a través de tutoriales, guías y manuales de las herramientas utilizadas para el desarrollo de este proyecto, y que habían sido definidas en una fase previa al inicio de este proyecto de

pasantía. Por ende, las actividades realizadas durante esta primera fase fueron las siguientes:

- Estudio de las herramientas involucradas en la solución.
- Estudio de la arquitectura definida para el proyecto.

### **3.2 Fase de Identificación.**

En esta fase se dio inicio a la definición de los servicios candidatos, así como los componentes y flujos a ser utilizados a lo largo de este proyecto de pasantía a través de la elaboración de una serie de documentos propios de SOMA® que sirvieron de guía a lo largo de la fase de identificación. A continuación se presentan las actividades realizadas:

- Estudio de las funcionalidades existentes en el sistema legado.
- Identificación de servicios candidatos.
- Elaboración de un Litmus test, cuestionario al cual se someten los servicios para determinar cuáles de los servicios candidatos eran factibles y cuáles no.
- Elaboración del portafolio de servicios.

### **3.3 Fase de Especificación**

En esta fase se inició el diseño de SOA-BPM. Aquí fue donde se diseñaron los componentes de forma detallada, y en donde además se determinó cuáles de los servicios eran factibles, generando una especificación de servicios para cada uno de ellos. Además, se generó una matriz de datos para cada uno de los servicios con la finalidad de poblar los atributos a ser implementados. Aunado a esto, se definió el flujo del proceso de solicitud de tarjetas de crédito a ser desarrollado posteriormente. Para esta fase, se realizaron las siguientes actividades:

- Creación de la matriz de datos de cada uno de los servicios.
- Elaboración de la especificación de servicios.

- Generación del documento de decisiones arquitectónicas del proyecto.
- Adaptación de los documentos ya elaborados e inclusión del resto de los servicios que representaban el alcance.
- Definición del flujo del proceso.

### **3.4 Fase de Realización e Implementación.**

En esta fase, en primer lugar, se validaron las decisiones de realización haciendo una exploración de viabilidad técnica para ejercitar las decisiones arquitectónicas y los factores de riesgo del proyecto a través de prototipos extensibles diseñados y desarrollados en las fases anteriores. Luego, durante la implementación, tanto los servicios como los procesos fueron desarrollados de acuerdo con las decisiones derivadas de la fase de realización de servicio. Finalmente, los servicios son orquestados y los flujos de mensaje son establecidos. Las actividades correspondientes son las siguientes:

- Desarrollo de un prototipo con la finalidad de mitigar riesgos y aclarar dudas técnicas de las herramientas así como de la comunicación con los sistemas legados.
- Definición del protocolo de intercambio de mensajes con el sistema legado.
- Integración del prototipo con el sistema legado.
- Elaboración del documento de cierre de la prueba de concepto.
- Desarrollo completo del proceso.

En el próximo capítulo se detallan los resultados de cada una de las fases de esta metodología para este proyecto de pasantía.

## CAPÍTULO IV

### 4. RESULTADOS OBTENIDOS EN BASE A SOMA.

En este capítulo se explicará de forma detallada todo el proceso de realización del proyecto de pasantía. El mismo se dividirá según las fases presentadas en el marco metodológico.

#### 4.1 Fase 1: Modelado

Las primeras semanas de la pasantía fueron dedicadas a entender los requerimientos y obtener un primer acercamiento a las herramientas que serían utilizadas. A continuación se explica detalladamente las actividades realizadas.

##### 4.1.1 Estudio de las herramientas involucradas en la solución.

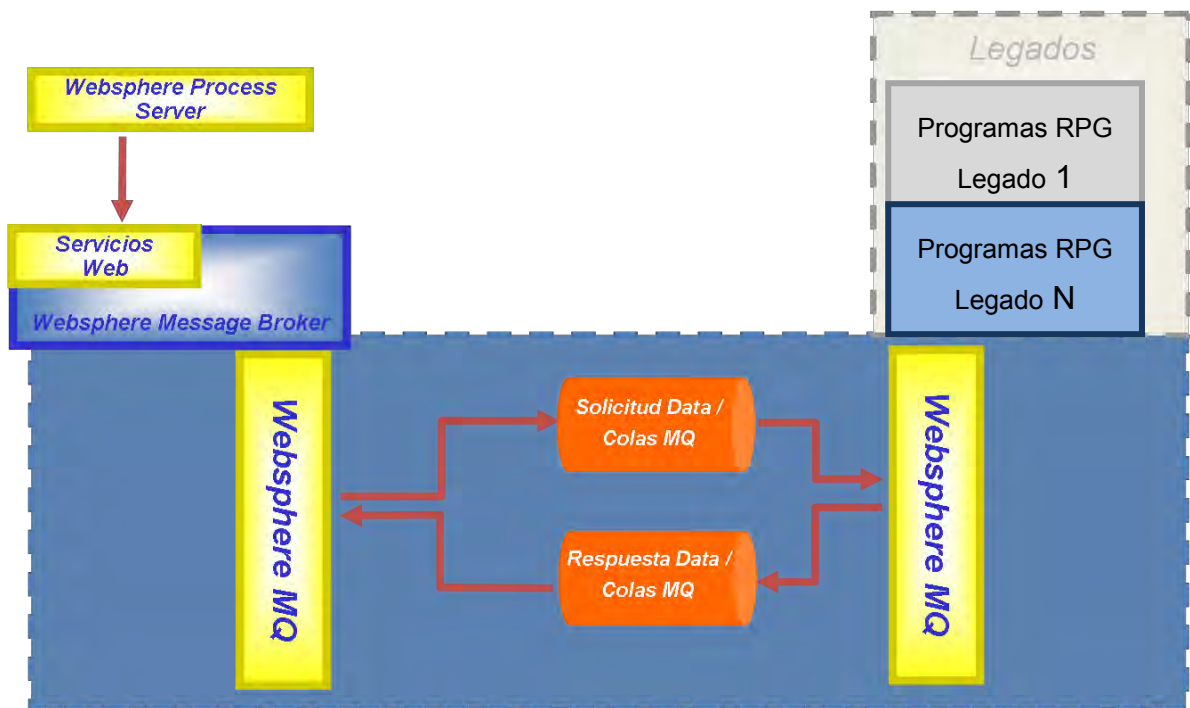
Como se mencionó en el capítulo 2, en una fase previa al inicio de la pasantía se definió el conjunto de herramientas a ser utilizados para desarrollar la solución. Las herramientas utilizadas se describen brevemente a continuación:

- **Desarrollador de Integración de WebSphere (WID):** Es una herramienta basada en Eclipse utilizada para elaborar el flujo del proceso y almacenarlo en el Websphere Process Server.
- **Lenguaje de Ejecución de Procesos de Negocio (BPEL):** Es el lenguaje con el cual se implementó la mayor parte de los procesos en el WID. Tiene la particularidad de que es del tipo *drag&drop*, es decir, la forma de programar en el es arrastrando nodos a un *canvas* y conectándolos para darles algún tipo de lógica.
- **Java:** Este lenguaje fue utilizado para desarrollar algunos de los procesos del WID, así como parte de la lógica del flujo en el WebSphere Message Broker Toolkit.
- **Espacio de Negocio de WebSphere (WebSphere Business Space, en inglés):** Es el programa en donde se maneja la interfaz gráfica. En esta herramienta, aquellos que utilizan el sistema por parte de la entidad financiera pueden ver su bandeja de tareas para saber cuáles de ellas están pendientes y así poder reclamarlas y trabajar en ellas. El Espacio de Negocio de WebSphere puede ser iniciado desde el WID para efecto de las pruebas que deben ser realizadas a la hora de hacer el desarrollo.

- **Websphere Message Broker Toolkit:** Es una herramienta basada en Eclipse con la cual se desarrollan los flujos de ejecución dentro del Websphere Message Broker, y donde además se manipulan los mensajes de manera que sean entendidos tanto por el Process Server como por los sistemas legados del banco. Además, la herramienta ofrece la posibilidad de probar los flujos ahí implementados, ayudando así al correcto desarrollo de cada uno de dichos flujos.
- **ESQL:** Es un lenguaje basado en SQL, con el cual se maneja parte de la lógica que hay dentro del bróker a la hora de desarrollar el flujo de mediación entre el servidor de procesos y el sistema legado de la entidad bancaria.
- **Explorador de Coreografías de los Procesos de negocio (Business Process Choreographer Explorer, por sus siglas en inglés):** El Explorador de Coreografías de los Procesos de Negocio es una funcionalidad que se encuentra dentro del Servidor de Procesos que permite verificar, luego de iniciada una instancia del proceso, en que parte del proceso se encuentra la ejecución. Esta herramienta se utilizó solamente para propósitos de prueba a la hora de la fase de Realización y Desarrollo y tuvo gran importancia a la hora de depurar errores, ya que a través de ella, se podía saber en que parte del proceso se detuvo la ejecución cuando había un error.

#### 4.1.2 Estudio de la arquitectura definida para el proyecto.

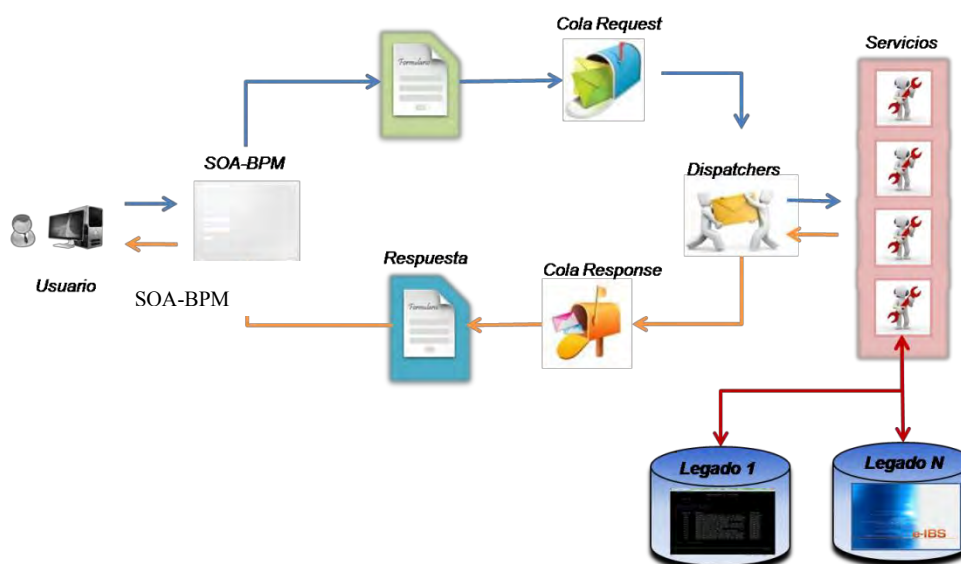
En la figura 4.1 se aprecian los elementos involucrados en la arquitectura de la solución SOA. En primer lugar, se encuentra el Servidor de Procesos de WebSphere, que es el servidor en contacto con la interfaz y es en donde está almacenado el proceso como tal. Entonces, cuando una información es requerida, se inicia el proceso. Desde el proceso son llamados una serie de servicios expuestos desde el servidor de Websphere Message Broker, que funge como orquestador de estos y además como mediador entre el Process Server y los sistemas legados de la entidad financiera.



**Figura 4.1.** Elementos involucrados en la arquitectura de la solución (SOA-BPM). (Elaboración Propia)

Cuando un servicio que se encuentra en el Websphere Message Broker es llamado, envía una trama a una cola de solicitud Websphere MQ, en donde se almacenan y son tomadas por el sistema legado del banco todas las solicitudes de ese servicio a través de un programa hecho en el llamado “Dispatcher”. El “Dispatcher” resolverá la solicitud y enviará la respuesta a una cola de respuesta, donde será tomada por el WebSphere Message broker para traducir la trama recibida en un XML que pueda ser leído por el Servidor de Procesos. Finalmente, el WPS es el encargado de hacer llegar la respuesta al usuario. En la figura 4.2 se puede ver a más alto nivel el diseño de la solución.





**Figura 4.2.** Diseño de la solución a alto nivel. (Elaboración Propia)

## 4.2 Fase 2: Identificación.

A continuación, se dará una descripción detallada de cada una de las actividades realizadas durante la fase de Identificación.

### 4.2.1 Estudio de la funcionalidad existente en el sistema legado.

En primer lugar, se procedió a estudiar las características del sistema utilizado hasta los momentos por la entidad financiera para la solicitud de tarjetas de crédito. De esta manera, se obtuvo una visión más clara del proceso deseado, así como también, de las funcionalidades con las que contaba para saber cuáles de ellas podían ser reutilizadas y cuáles debían ser agregadas con la finalidad de optimizar el proceso. La reutilización es uno de los factores importantes de SOA-BPM, ya que representa una mejora en cuanto al tiempo de desarrollo y de la calidad del sistema, ya que permite obtener funciones específicas que son relevantes.

El cliente contaba con un sistema para el proceso de tarjetas de crédito en el cual los funcionarios interactuaban directamente con una serie de programas hechos en el lenguaje RPG que se encontraban en los sistemas legados. Debido a la compleja que resulta la usabilidad de éstos sistemas, el uso de éstos retardaba y complicaba el proceso. Adicionalmente, la información requerida no estaba centralizada, por lo tanto, tomando en cuenta que una solicitud de tarjeta de crédito almacena información en diferentes sistemas legados, el analista debía interactuar directamente con cada uno de los programas para obtener

la información requerida. Finalmente, se concluyó que se debía desarrollar un *middleware* que interactuara con los programas RPG del sistema legado y que a la vez proporcionara buen desempeño y una interfaz amigable para facilitar a todos los actores el proceso de solicitud de tarjeta de crédito.

#### 4.2.2 Identificación de servicios candidatos.

Una vez obtenidos los casos de uso y habiendo analizado el sistema actual, se listó una serie de servicios candidatos que podrían ser incluidos en la solución.

Con el objetivo de optimizar el portafolio de servicio, cada servicio fue analizado mediante una prueba llamada *Litmus Test*. En función de una serie de criterios tanto técnicos como funcionales se identificó cuales servicios debían ser expuestos.

Los criterios basados en las mejores prácticas de la industria se describen a continuación<sup>[11]</sup>:

- a) **Alineación al negocio:** Evalúa el nivel de correspondencia entre el negocio y el servicio, basado en los siguientes puntos:
  - Funcionalidad empresarial de soporte a los procesos y objetivos empresariales.
  - Intención de financiar el servicio en todo su ciclo de vida.
  - Uso y exposición del servicio hacia socios externos y/o potenciales clientes.
  - Existencia actual y en el futuro de reutilización del servicio.
- b) **Servicio Componible:** Determina la posibilidad que el servicio participe en la composición de otros servicios los cuales son determinados por los siguientes puntos:
  - El servicio soporta los requerimientos no funcionales.
  - El servicio es sin estado.
  - El servicio es independiente y puede cumplir por si solo un objetivo empresarial.
  - El servicio puede cooperar con otros servicios para cumplir objetivos empresariales.

- La tecnología de implementación del servicio es neutral.
- c) **Descripción:** Evalúa las capacidades del servicio de poder auto describirse, basados en los siguientes puntos:
- El servicio posee una descripción externa la cual es distinta e independiente de la implementación física subyacente.
  - El servicio puede ser descubierto y enlazado con la descripción del mismo.
  - Contiene la descripción del servicio metadatos que permite la identificación del mismo.
- d) **Reusabilidad:** Evalúa las capacidades del que el servicio pueda ser re usado por diferentes procesos de negocio.
- e) **Viabilidad Técnica:** La viabilidad técnica asegura que el servicio se pueda implementar y desplegar, basado en los siguientes puntos:
- El esfuerzo de implementación y gestión del servicio es razonable y fácilmente viable.
  - El servicio puede ser implementado por un número mínimo de sistema proveedores.
  - No existen retos tecnológicos explícitos conocidos ni anticipados.

De esta forma, se obtuvo una serie de servicios que fueron los considerados para el desarrollo de la solución (Apéndice A).

#### 4.2.3 Elaboración del portafolio de servicios.

Una vez obtenidos los servicios que resultaron factibles, se procedió a realizar el documento llamado portafolio de servicios. En éste, para cada servicio identificado se especificaron los siguientes puntos:

- **Descripción:** Una breve especificación del significado de cada servicio.
- **Propietario del servicio:** Entidad a la que pertenece el desarrollo de servicio.

- **Proceso de Negocio:** Proceso al que pertenece el servicio. Al ser un solo proceso de negocio el que se iba a desarrollar, todos los servicios pertenecían al proceso de Solicitud de Tarjetas de Crédito (Solicitud TDC).

- **Componente del negocio/Funcionalidad del Negocio:** Indica qué actor da uso a cada servicio.

- **Activo asociado al servicio:** Especifica a cuál de los sistemas legados va asociado el servicio.

En el Apéndice C se encuentra un modelo de portafolio de servicio para uno de los servicios.

### 4.3 Fase 3: Especificación.

A continuación, se dará una descripción detallada de cada una de las actividades realizadas durante la fase de Identificación.

#### 4.3.1 Elaboración de la Especificación de servicios.

En el inicio de esta fase y, una vez terminado el portafolio de servicios, se elaboró un documento que contenía de forma más detallada la información correspondiente a cada uno de los servicios llamado Especificación de Servicios (Apéndice D).

En una primera etapa, se incluyeron en el documento los siguientes puntos:

- **Operaciones del servicio:** Esta sección del documento contenía una descripción de las diferentes operaciones que contiene el servicio para exponer su funcionalidad.

- **Eventos:** Descripción de los eventos del proceso de negocio que desencadenan las operaciones del servicio.

- **Correspondencia Servicio-Componentes:** Mapeo sobre cuáles componentes en la arquitectura de la aplicación debían contener la implementación del servicio descrito en dicho documento.

- **Decisiones de Arquitectura para la realización del servicio.**

En una segunda etapa, una vez obtenida la retroalimentación por parte del cliente sobre cómo sería la estructura de mensajes de entrada y salida de cada servicio, se refinaron las

especificaciones de servicio y se agregaron los siguientes parámetros:

- **Atributos relacionados con el contexto:** Descripción de los atributos en el contexto de la ejecución de un servicio que pudieran afectar el flujo de los mensajes del servicio o el ensamblado de múltiples mensajes.
- **Reglas y políticas:** Reglas y políticas generales que aplicaban al desarrollo de cada servicio.
- **Diagrama de contexto de servicios:** Descripción del contexto de todas las entidades que interactúan con el servicio descrito en el documento.
- **Diagrama de Clases de mensajes de entrada y de salida:** Descripción de los mensajes de entrada y salida para las diferentes operaciones del servicio
- **Mensajes del Servicio:** Mensajes de Error, Informativos o de Advertencia que se pudieran dar en cada uno de los servicios.
- **Entradas y salidas del programa RPG:** Definición de las tramas de entrada y salida para los programas RPG que responderían a las solicitudes realizadas por este servicio.
- **Requerimientos funcionales y no funcionales de servicios.**

Cabe destacar que se realizó un documento de especificación de servicio para cada uno de los servicios realizados (Ver servicios en Apéndice A).

#### 4.3.2 Creación de la matriz de datos de cada uno de los servicios.

Para cada uno de los servicios realizados, se generó, junto con el personal de la entidad bancaria, una matriz de datos en donde se representaba para cada uno de los atributos que tenía la entrada y la salida de cada servicio, el tipo de dato, la longitud y el formato como tal para establecer una comunicación efectiva del sistema con los sistemas legados de la empresa.

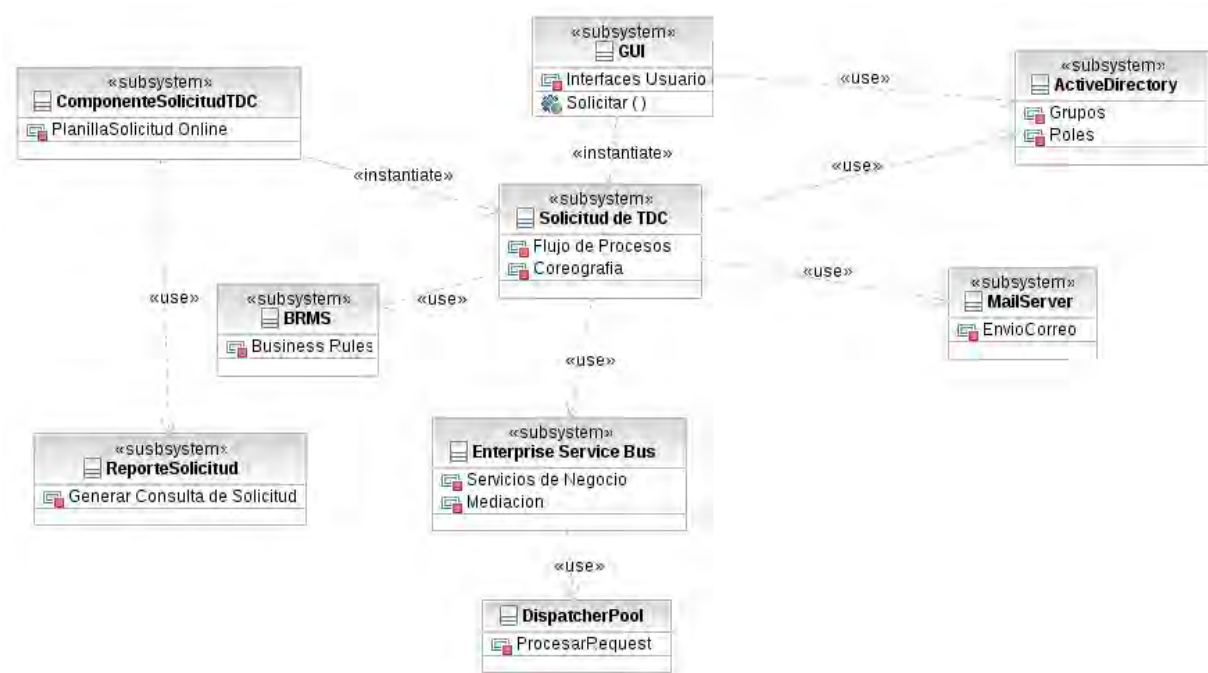
El trabajo descrito anteriormente se hizo, en primera instancia, para los servicios de la prueba de concepto, con el fin de hacer pruebas de comunicación con el sistema legado. Al culminar la prueba de concepto, se creó una matriz de datos para el resto de los servicios .



trazabilidad y justificación para su inclusión en la solución, concepto clave en la metodología SOMA.

Un componente de servicio es la realización de un subsistema, una agrupación lógica de servicios alineados funcionalmente que tienen suficiente importancia para el negocio. El modelo documentó los componentes de servicios, componentes funcionales, componentes técnicos y subsistemas que realizaron los aspectos funcionales de la solución.

En la figura 4.3 se puede apreciar la división de los subsistemas del proyecto y la relación que guardan. En el Apéndice B se puede encontrar la descripción de cada uno de los componentes.



**Figura 4.4.** Subsistemas de la solución.

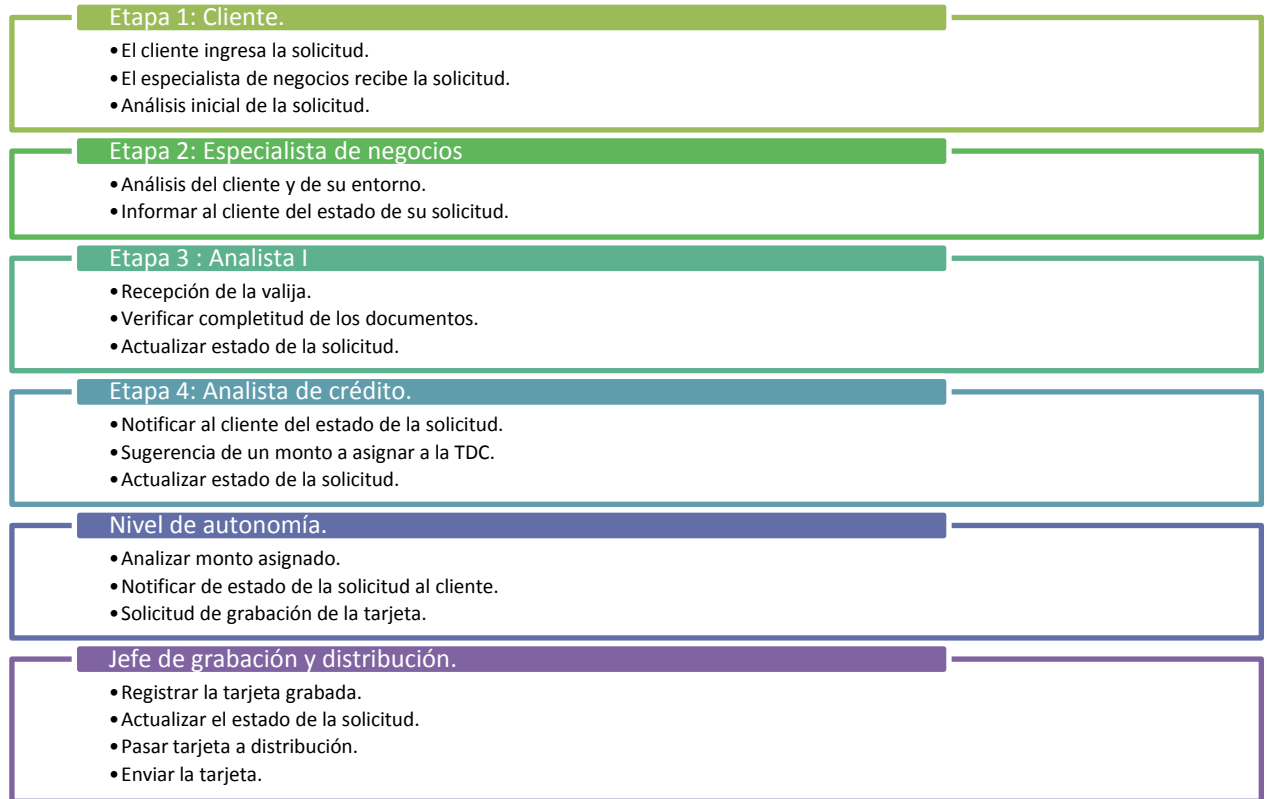
Seguido a esto, para cada subsistema se asoció una serie de componentes de acuerdo a la función que cumplía cada uno.

#### 4.3.4 Definición del proceso a utilizar en el desarrollo.

Con la información obtenida hasta el momento, se realizó el diseño del proceso a utilizar

para el desarrollo del proyecto. En la figura 4.4, se muestra un sumario de las etapas por las cuales es llevado a cabo el proceso.

¿



**Figura 4.5.** Etapas del proceso de solicitud de TDC.

A lo largo del proceso de solicitud de tarjetas de crédito, participan una serie de actores, cada uno de los cuales desempeña una función específica en el sistema. A continuación se explica detalladamente el papel que desempeña cada actor y cómo éste representa un número de funciones dentro del diagrama del proceso.

### **Etapa I: El cliente y Especialista de negocios.**

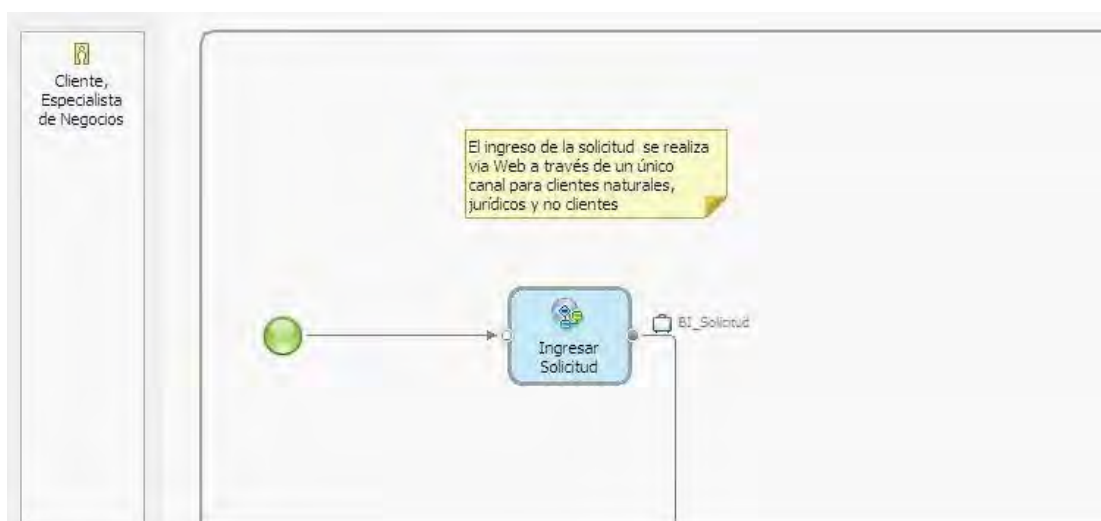
En la fase inicial del proceso, el cliente ingresa una solicitud de tarjeta de crédito vía web. Al hacer esta solicitud, se inicia el proceso, en donde el especialista de negocios recibe en su bandeja de tareas una nueva solicitud pendiente. Al recibir la solicitud, el especialista debe



esperar que el cliente acuda a la agencia a entregar los recaudos solicitados.

Una vez obtenidos los recaudos, se analiza la solicitud y se consulta si el cliente es recurrente. En caso positivo, simplemente se analizan los recaudos obtenidos y si están completos. En caso de faltar algún recaudo, se le notifica al cliente y se repite el proceso. Finalmente, se registra la solicitud en el sistema legado y se pasa a la siguiente etapa.

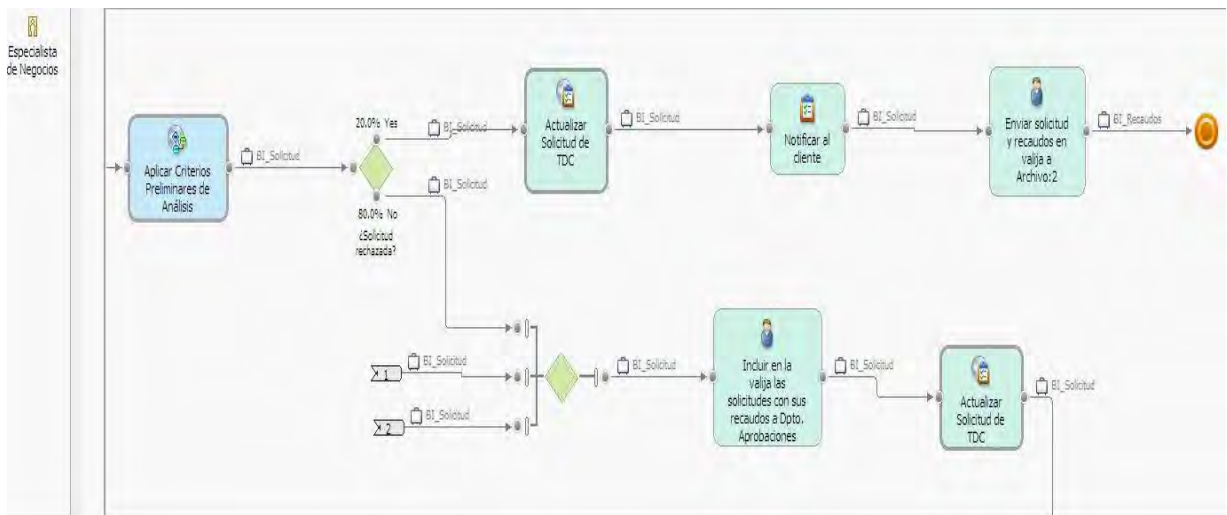
En caso contrario, el especialista de negocios verifica si es un cliente natural o jurídico, y según sea el caso, se establece la lógica correspondiente de análisis a cada uno de los casos, luego se verifica la completitud de los datos y se registra en el sistema legado. En la figura 4.5 se puede apreciar la fase inicial de éste proceso a alto nivel.



**Figura 4.6.** Diagrama de modelado de alto nivel del ingreso de la solicitud por parte del cliente.

## **Etapas II: Especialista de negocios.**

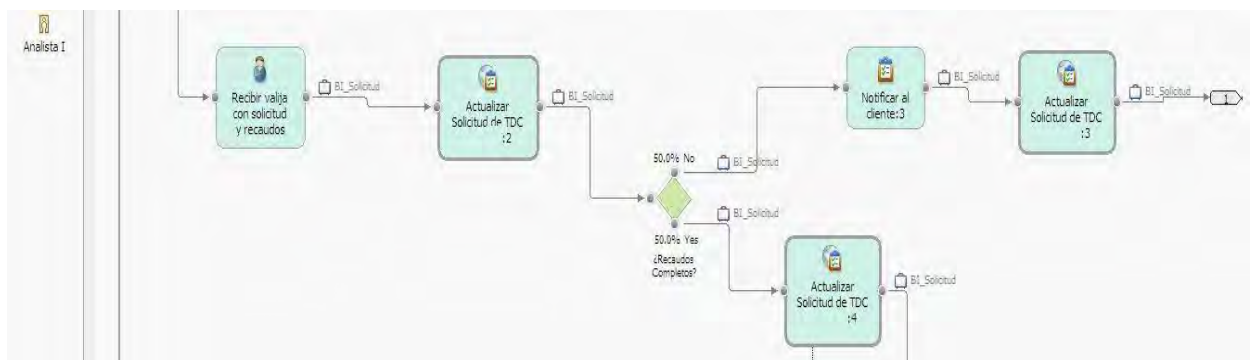
En esta etapa, el especialista de negocios aplica unos criterios preliminares a la solicitud enviada por el cliente, es decir, hace un análisis del cliente así como de su entorno. De ser rechazada la solicitud, el cliente será notificado de la situación. En caso contrario, de igual forma se informa al cliente que su solicitud está pre-aprobada y se incluyen, tanto la solicitud como los recaudos, en una valija que será recibida por el Analista I. En la figura 4.6 se puede apreciar el diagrama de modelado de alto nivel de dicha etapa.



**Figura 4.7.** Diagrama de modelado de alto nivel de las operaciones realizadas por el Especialista de negocios.

### **Etapa III: Analista I.**

En esta etapa, el Analista I recibe la valija, y actualiza el estado de la solicitud según la completitud de los recaudos. En caso de faltar alguno, se notifica al cliente y se envía la solicitud nuevamente al especialista de negocios para que éste reciba los recaudos faltantes y los coloque en la valija y los procese de nuevo al Analista I. En caso de estar completos, se actualiza la solicitud de TDC y se continúa con el proceso. En la figura 4.7, se puede apreciar el segmento del proceso correspondiente al Analista I.

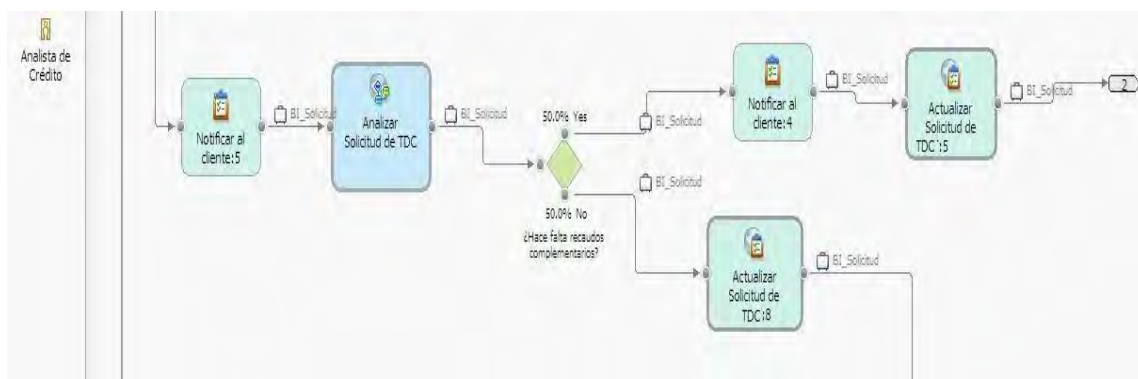


**Figura 4.8.** Diagrama de modelado de alto nivel de la etapa 3 de la solicitud de TDC. Analista I.

### **Etapa IV: Analista de Crédito.**

Una vez el Analista de crédito recibe la solicitud, notifica al cliente sobre estado de la

misma y procede a analizarla y decidir si hace falta algún recaudo complementario. En caso afirmativo, se indica al cliente que son requeridos recaudos adicionales, envía la solicitud nuevamente al especialista de negocios para que reciba los recaudos faltantes y se repite el proceso, en caso contrario se continúa con el proceso, en donde el analista sugiere un monto según su análisis y actualiza la solicitud. En la figura 4.8 se puede observar el detalle de este segmento del proceso.

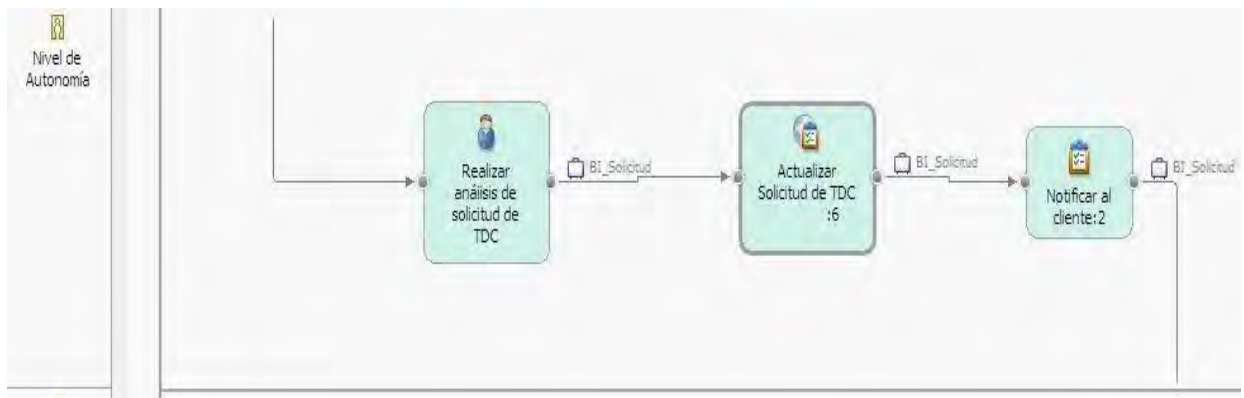


**Figura 4.9.** Diagrama de modelado de alto nivel de la etapa 4 de la solicitud de TDC. Analista de crédito.

### **Etapla V: Nivel de Autonomía.**

En esta etapa, una vez aprobada la solicitud por el analista de crédito, se analiza el monto asignado y se notifica al cliente que su tarjeta pasó a ser grabada, es decir, el plástico será elaborado y troquelado con el nombre del cliente para luego ser entregado o a la agencia receptora o al cliente en el domicilio que éste refiriera para su entrega

. En la imagen 4.9 se puede apreciar el segmento del proceso correspondiente al nivel de autonomía.

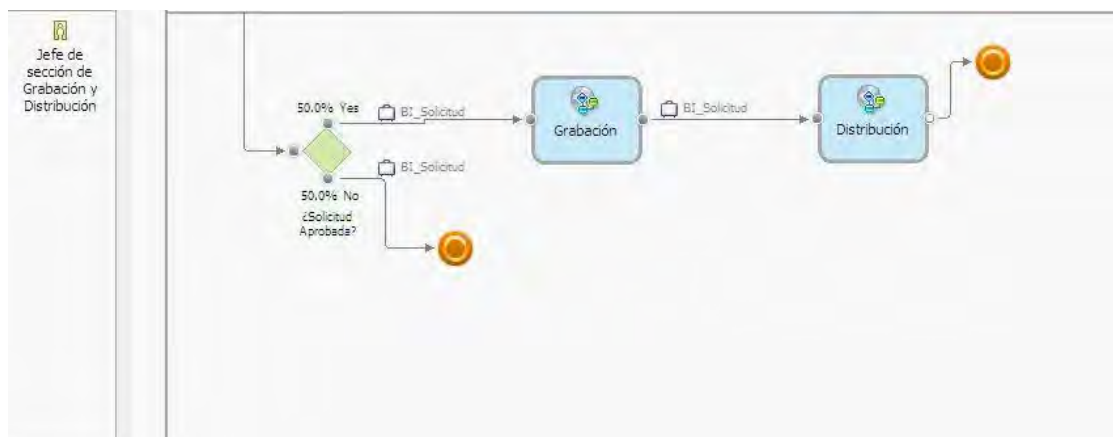


**Figura 4.10.** Diagrama de modelado de alto nivel de la etapa 5 de la Solicitud de TDC. Nivel de autonomía.

### **Etapla VI: Jefe de sección de Grabación y Distribución.**

Esta etapa solo es posible si la solicitud de tarjeta de crédito fue aprobada. En dicha etapa, se debe registrar la tarjeta grabada, actualizar el estado de la solicitud y finalmente distribución, en donde se hace el envío de la tarjeta. De ser recibida la tarjeta por la agencia, el especialista de negocio la registrará en el sistema, mientras que, en el caso de ser entregada o devuelta por el *Courier* (Servicio de envíos), será el mismo Jefe de Sección de Grabación y Distribución quien deberá registrar la entrega o devolución y actualizar la solicitud, para luego finalizar el proceso.

En la figura 4.10 se puede apreciar el diagrama del proceso a nivel del Jefe de sección de grabación y distribución.



**Figura 4.11.** Diagrama de modelado de alto nivel de la etapa 6 de la solicitud de TDC. Grabación y Distribución.

#### 4.4 Fase 4: Realización e Implementación.

Para el desarrollo del sistema, en primera instancia, se realizó una prueba de concepto para mitigar riesgos a la hora de realizar la implementación completa del sistema y, en segunda instancia se realizó el desarrollo del proceso completo de solicitud de TDC. A continuación, se describen cada una de las actividades realizadas durante esta fase:

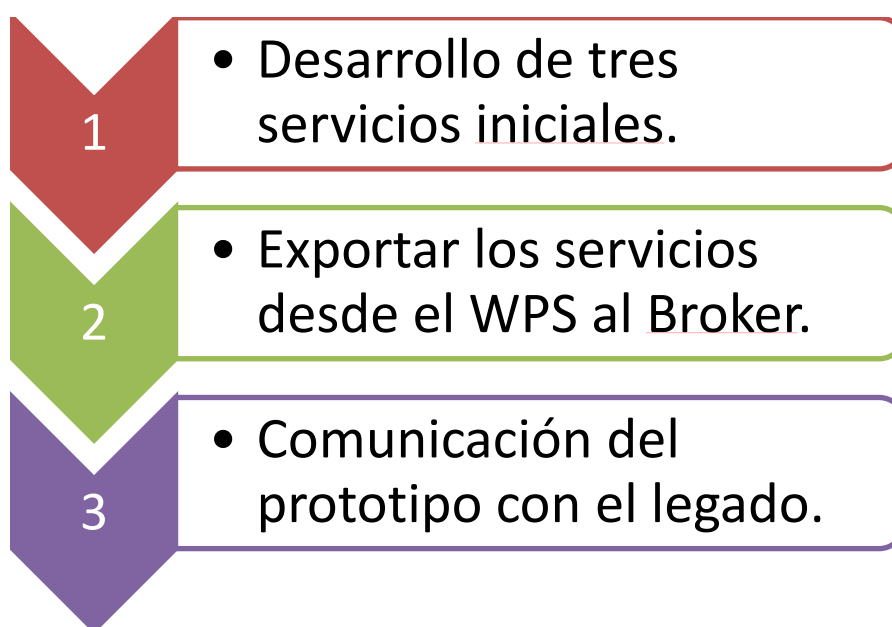
##### 4.4.1 Desarrollo del prototipo para la Prueba de Concepto.

Para ejecutar la prueba de concepto, se realizó un prototipo que inicialmente contemplaba tres servicios:

- **Consulta de Cliente:** Se envía un número de cédula al sistema legado y se obtiene un objeto de negocio con la información correspondiente al cliente solicitado.
- **Consulta de una Tarjeta de Crédito:** En este servicio se envía un número de tarjeta de crédito y se obtiene un objeto de negocio con los datos correspondientes a la misma.
- **Registro de una solicitud:** Servicio que envía un objeto de negocio con los datos de la solicitud a registrar y hace el registro del sistema legado. Recibe un mensaje confirmando el éxito de la operación.

Finalmente, luego de terminados estos servicios, se agregó un servicio adicional para probar el manejo de estructuras con registros múltiples que consistía en **Consultar lista de tarjetas de un cliente**. En éste, se ingresaba el número de identificación del cliente y se devolvía una lista de todas las tarjetas que tenía el mismo.

En base a esto se dividió la prueba de concepto en tres etapas (figura 4.11):



**Figura 4.12.** Etapas del desarrollo de la Prueba de Concepto.

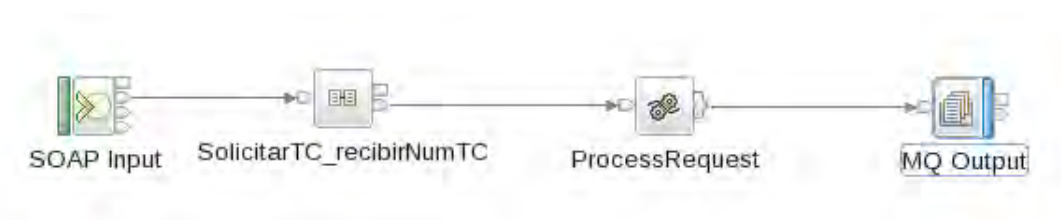
- En una primera etapa, se desarrollaron los tres servicios iniciales a nivel del WPS, la respuesta de los servicios fue realizada en la implementación del proceso, ya que aún no había comunicación ni con el ESB, ni con el sistema legado. La finalidad de esto fue el diseño y la correcta elaboración del proceso a nivel del servidor de procesos.
- La segunda etapa fue exportar los servicios desde el WPS al WebSphere Message Broker que, como se mencionó anteriormente, funge como Bus de Servicios Empresariales y comunica el Process Server con los sistemas legados, mediante un protocolo de comunicación establecido. En esta etapa, se bajó de nivel los servicios, es decir, se colocaron a nivel del Broker, desde donde eran llamados por el Process Server. Para simular la comunicación con el sistema legado, similar a como se hizo en la primera etapa, se generaban mensajes desde el mismo Message Broker que respondían a solicitudes enviadas desde el WPS. La finalidad de esto fue establecer el flujo de mediación entre el WPS y el sistema legado para tener una base robusta al momento de la comunicación definitiva.
- Finalmente, en una última etapa, se comunicó el resultado obtenido de las dos etapas anteriores con el sistema legado del banco. Para ello, en primer lugar se sustituyó en el desarrollo del flujo de mediación aquellas porciones de la implementación en donde se

generaban mensajes predefinidos, por la lógica necesaria para establecer la comunicación con el sistema legado.

#### 4.4.2 Definición del protocolo de intercambio de mensajes con el sistema legado.

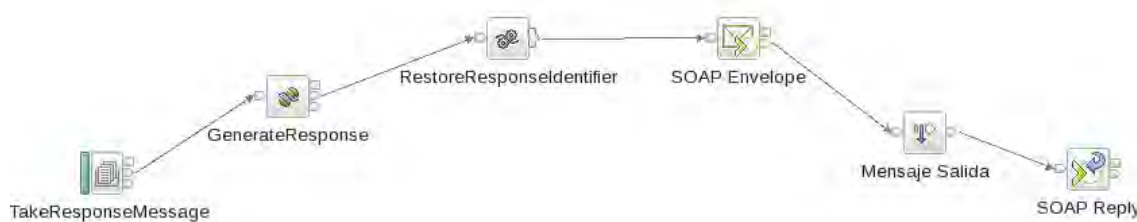
Para establecer una comunicación efectiva entre el Message Broker y los “*dispatcher*” del sistema legado, se estableció un protocolo de comunicación en conjunto con el personal de la entidad financiera. Se acordó que el sistema legado enviaría y recibiría tramas de datos basándose en la información obtenida de la matriz de datos de cada uno de los servicios, en donde se encontraba especificado el tamaño de cada atributo. Hablar de trama de datos implica que se envía la información correspondiente en una sola cadena en un orden específico, desde la cual, se extrae la información basándose en el tamaño de cada atributo. A nivel de Message Broker, al recibir un mensaje del Process Server en forma de XML, se procesa, se extrae la información del mensaje y se envía al sistema legado en forma de trama de datos. De igual forma, cuando llega una respuesta del sistema legado, a nivel del Broker se extrae dicha información, se separa y se construye el XML que sería recibido por el Process Server.

En la figura 4.12, se puede apreciar el flujo de solicitud de uno de los servicios. En primer lugar, el nodo SOAP Input espera que el servicio sea llamado y por ende la solicitud recibida para iniciar el flujo. Al recibir la solicitud, se desencadena el proceso, y el mensaje pasa al nodo llamado *SolicitarTC\_recibirNumTC*, el cual remueve los encabezados SOAP que trae el mensaje. Luego, en el *ProcessRequest* es donde se convierte el mensaje de un XML a una trama basada en la lógica definida y finalmente el mensaje es colocado en su cola correspondiente utilizando el MQ Output.



**Figura 4.13.** Flujo de solicitud a nivel del *broker*.

Una vez encolado, el mensaje es recibido por el “*Dispatcher*” en el sistema legado, el cual busca la información que sea necesaria y entonces dirige el mensaje a la cola de respuesta. Al ocurrir esto, se activa el flujo de respuesta (figura 4.13), en donde el nodo “*TakeResponseMessage*” se encarga de desencolar el mensaje. Es aquí, cuando el mensaje pasa por el nodo “*GenerateResponse*”, que recibe la trama y arma en base a ella el XML que será enviado al Servidor de Procesos. Finalmente, el nodo “*RestoreResponseIdentifier*” le asigna un identificador de respuesta al mensaje, para luego obtener el encabezado SOAP y ser enviado con el “*SOAP Reply*” para el Servidor de Procesos.



**Figura 4.14.** Flujo de respuesta a nivel del *broker*.

#### 4.4.3 Desarrollo completo del proceso.

Para obtener el diagrama de ensamblaje inicial del proceso, se exportó el modelo obtenido del WBM y se importó en la herramienta de integración y desarrollo (WID). Una vez recibido, se le agregó la lógica al mismo, ya que al exportarlo se genera simplemente un esqueleto del proceso sin la lógica y los servicios desarrollados.

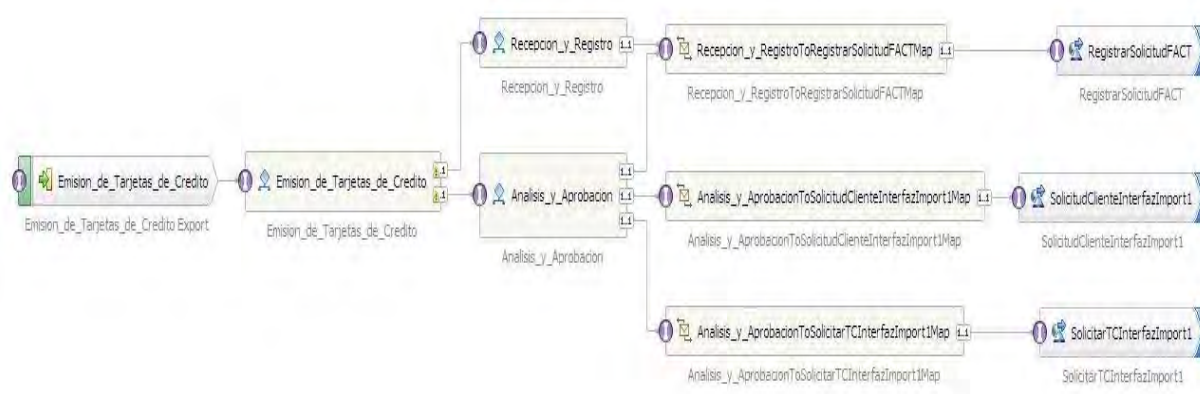
Para hacer el desarrollo del proceso, se creó en primer lugar un formulario a fin de hacer una simulación completa. Dicho formulario no representa el definitivo, ya que el desarrollo de éste no está contemplado en el plan de proyecto, y por ende es responsabilidad de la entidad bancaria.

Para hacer el formulario, se creó un cliente de servicio web a partir del WSDL obtenido al generar el servicio que expondría el proceso completo. Dicho servicio tiene una interfaz



que recibe una solicitud de TDC y devolvía una cadena de caracteres (*String*) con un mensaje de confirmación. Al generar dicho cliente, se creó la base del formulario en un archivo JSP (Java Server Pages), es decir, se generó un archivo en donde se podía ingresar toda la información de una solicitud demarcada por una serie de campos y que entonces, al completar toda la información, se enviaba el formulario y, con esta información, se poblabla el objeto de negocio que representaba la solicitud, dando inicio al proceso a nivel del WMB. Adicionalmente, se añadió a dicho formulario aquellos servicios que consultaban información en el sistema legado para ser mostrada en el mismo (por ejemplo, “Consultar Estado”, en donde se consulta en el sistema legado todos los estados del país para ser mostrados a modo de “ComboBox” en el formulario).

Posteriormente, para iniciar el proceso en el Message Broker, se utilizó el API del mismo creando un nodo que recibe la solicitud y que en él, mediante funciones predeterminadas del Broker, se inicia una instancia del proceso y se continúa con el flujo. Para comunicarse con el Process server, al final de este servicio, hay un SOAP *output* que envía el objeto de negocio poblado hacia el WPS a través de un servicio web. Una vez en el WPS, se da inicio al proceso (Figura 4.14).



**Figura 4.15.** Diagrama de ensamblaje del proceso a nivel del WPS.

En el WPS, se observa el nodo “Emission\_de\_tarjeta\_de\_credito Export”, encargado de exponer todo el proceso completo para ser tomado por vía de servicios web. Por otro lado, también se encuentran un proceso y dos subprocesos descritos a

continuación:

- **Emisión\_de\_tarjetas\_de\_credito:** Es el proceso macro, en él simplemente se orquestan los otros dos subprocesos para darles un orden coherente de ejecución.
- **Recepción y Registro:** En este sub-proceso, se recibe el formulario y se registra en el sistema legado del banco en primera instancia, es decir, la parte inicial del servicio descrito.
- **Análisis y aprobación:** En este sub-proceso se hace el grueso de la lógica de la Solicitud de TDC, desde analizar los recaudos, hasta grabación y distribución. Desde este proceso, se llaman los servicios que están expuestos en el Broker mediante un nodo llamado “Import” En el cual se hace un enlace especificando el *endpoint* en donde se encuentran alojados los servicios.

Finalmente, a nivel del Broker, se desarrolló el flujo de mediación de cada uno de los servicios tal y como se explicó en la sección 4.6.2.

A lo largo de todo el desarrollo se hicieron pruebas tanto internas como en conjunto con el personal del banco. En primera instancia, al no haber conexión con los legados se simulaba el flujo de mensajes con las herramientas de prueba que provee WebSphere Integration Developer y se seguía el flujo de dichos mensajes para detectar errores. Una vez establecida la conexión con el sistema legado, se realizaron pruebas en conjunto con el personal técnico del banco en las cuáles el sistema recibía una trama de datos de parte del sistema legado y la manejaba y retornaba la respuesta, y de igual forma se enviaba un XML al legado del banco para determinar que el intercambio de mensajes se estuviera llevando de la mejor forma.

#### 4.5 Fase 6: Despliegue.

Adicionalmente, se planea la fase de despliegue de los procesos modelados en este proyecto de pasantía. Para ello, se deben seguir los pasos descritos a continuación:

1. Se realizará el despliegue del sistema en los servidores finales de la Entidad Financiera.
2. Se deben ejecutar casos de prueba una vez el sistema sea implantado en los servidores del banco, tanto a nivel de usuario (cliente del banco) como a nivel de los analistas y especialistas de negocio.
3. Realizar el manual del sistema a ser consultado por el cliente y usado para el debido entrenamiento del personal que laborará bajo esta nueva herramienta

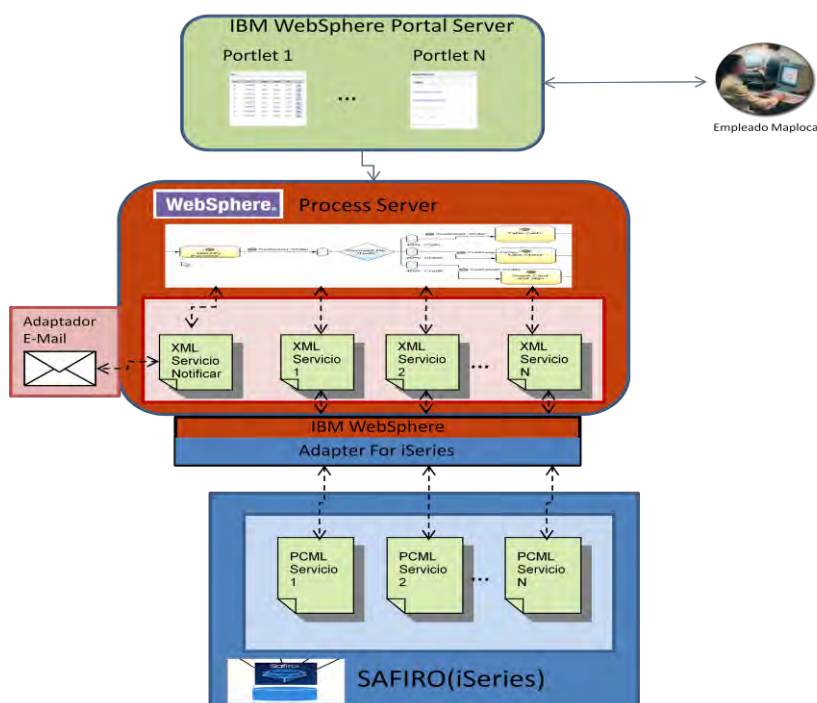
#### 4.6 Logros Adicionales: Elaboración de Portlets para un Módulo de Consulta de Requisiciones de Compra.

Una vez finalizado el desarrollo en el proyecto del módulo de Solicitud de Tarjetas de Crédito, se participó de un proyecto adicional que consistía en la elaboración de *portlets* para un módulo de Consulta de Requisiciones de Compras para una empresa en el sector acerero. Una Requisición de Compra es una autorización del departamento de compras para abastecer a la compañía de bienes o servicios que no se encuentran en inventario o están por agotarse. Ésta a su vez es originada y aprobada por el departamento que requiere los bienes o servicios.

Los *portlets* son componentes web diseñados específicamente para ser agregados en el contexto de una página compuesta. Usualmente, muchos *portlets* son invocados en la solicitud de una página portal, aplicación web que comúnmente provee personalización e integración de contenido desde diferentes fuentes y almacena la capa de presentación de los sistemas de información.

La arquitectura de este proyecto (Figura 4.15), como la del módulo de solicitud de Tarjetas de Crédito, está igualmente orientada a servicios, con la diferencia de que, en este caso, se empleó un adaptador llamado “Adaptador para iSeries de WebSphere” en

sustitución del Broker. Este adaptador se expone desde el servidor de procesos y comunica con el sistema legado haciendo un mapeo entre el XML del proceso y una serie de archivos, creados bajo el lenguaje de códigos de llamadas a programas (PCML), sirviendo de mediador entre el Servidor de Procesos y el sistema legado.



**Figura 4.16.** Arquitectura del proyecto de la empresa del sector acerero (elaboración propia).

Adicionalmente, la capa de presentación, en vez de ser manejada a través del Business Space, utilizó *portlets* a través de un servidor llamado WebSphere Portal Server, en el cual fueron almacenados.

Para la realización del *portlet* correspondiente a la consulta de una requisición de Compra, se utilizó el lenguaje Java, a través del *framework* JSF (Java Server Faces) y en donde además, se utilizó el *ajax* que se encuentra dentro de las librerías de JSF de IBM para hacer las actualizaciones de ciertas áreas de los *portlets* sin necesidad de actualizar toda la página.

Aunado a esto, para la comunicación del *portlet* con el servidor de procesos, se creó un cliente de servicios web dentro del proyecto del *portlet*. En este cliente se especificaba el *endpoint* o enlace donde se encontraba el servicio a nivel del servidor de procesos, y entonces se obtenía el WSDL de ese servicio. Al crear el cliente de

servicios web, automáticamente se generaba en el proyecto *portlet* una serie de clases de java que contenían operaciones que facilitaban la comunicación con el servicio web.

Para hacer dicha comunicación, simplemente hay que poblar con los datos necesarios, un Objeto que contenga los mismos atributos que los definidos en la entrada de la interfaz asociada al servicio, y luego llamar un método que establece la comunicación y devuelve la información de respuesta en otro Objeto que representa el atributo de salida de la interfaz de dicho servicio. En la figura 4.16 se encuentra el *portlet* realizado.

The screenshot shows a web portlet titled "ConsultarRqc". It is divided into two main sections. The top section, "Datos de Búsqueda", contains a dropdown menu for "Cód. Compañía" with "SS" selected, a text input for "Número RQC", and a "Submit" button. The bottom section, "Información Requisición de Compras", displays the following details:

Número RQC:	273		
Compañía:	MAPLOCA - COMPAÑIA DE PRUEBA		
Sucursal:	01 CARACAS		
Unidad Solicitante:	1100 GCIA. DE INFORMÁTICA Y		
Descripción RQC:	Desar		
Fecha Elaboración:	2011-11-20	Fecha Requerida:	2011-12-25
Lugar de Entrega:	Caracas	Fecha Modificación:	Información no disponible
Proveedor:	Información no disponible		
Descripción Estado RQC:	POR APROB.		
Monto Total RQC:	0		

**Figura 0.17.** Portlet de consulta de una requisición de compra.

Este portlet se divide en 3 partes:

- El área de ingreso de datos de búsqueda de una requisición.
- Información general de la requisición de compras.
- Detalles de cada artículo de la requisición, presentados en una tabla.

Para el desarrollo de este proyecto, se implementarán una serie de portlets para mejorar 3 subprocesos del ciclo de egresos, los cuales se listan a continuación.

- Requisición de Compras.

- Orden de Compras.
- Recepción de Materiales.

Estos procesos contribuirán en gran parte a la automatización de muchas de las actividades actuales de la empresa, ya que actualmente muchas de sus actividades se realizan de forma manual.

## CONCLUSIONES Y RECOMENDACIONES

Al culminar el proyecto de pasantías, se desarrolló un *middleware* el cual sirve como orquestador de todas las actividades relacionadas al proceso de solicitud de Tarjetas de Crédito de una entidad bancaria de importancia a nivel nacional. A continuación, se presentan las conclusiones derivadas a la elaboración de este proyecto de pasantías así como las lecciones aprendidas:

- Al ser la metodología SOMA orientada a servicios, esta garantizó el correcto desempeño de los procesos y del sistema.
- La prueba de concepto fue un paso importante en el desarrollo del proyecto, ya que en ella se resolvió una serie de dificultades que fueron cruciales a la hora de hacer el desarrollo.
- La documentación adecuada permitió llevar un registro de todas las actividades realizadas durante todas las fases del proyecto. Este es un factor importante ya que debido a la dinámica del proyecto varias personas ingresaron al equipo una vez iniciado el proyecto, y esto sirvió de ayuda para su rápida adaptación.
- SOA-BPM facilitó la implementación de este proyecto, ya que los servicios y los procesos de negocio son de fácil implementación. En donde además, la comunicación con los sistemas legados es transparente al usuario, por lo tanto, el personal de la entidad bancaria no necesitan saber cómo se da ese intercambio de información, sino simplemente enviar los datos necesarios.
- Las herramientas WebSphere de IBM resultaron en una gran ventaja para el desarrollo, ya que tiene numerosas funcionalidades para hacer desarrollos en SOA, y en donde además, gran parte del desarrollo se hizo mediante una interfaz *drag&drop*, lo que redujo en gran medida la complejidad a la hora de desarrollar.
- Luego del desarrollo de este proyecto, el cliente del banco podrá crear una solicitud de tarjetas de crédito desde sus casas.

A continuación se presentan una serie de recomendaciones para futuros desarrollos posteriores a este proyecto, basados en la experiencia adquirida durante el modelado de procesos para un sistema de solicitud de TDC. Al seguir estas recomendaciones, se garantiza el desarrollo exitoso de cualquier proyecto posterior con condiciones óptimas y en el cuál se prevendrán situaciones que a veces generan problemas que tardan tiempo en solucionarse.

- A la hora de definir el protocolo de intercambio de mensajes con el sistema legado del cliente, se debe establecer la codificación a ser utilizada (*encoding*), ya que si esto no es tomado en cuenta, podría generar problemas de comunicación que son difíciles de detectar.
- Cuando se desee intercambiar atributos que contengan registros múltiples con los sistemas legados del cliente, se debe colocar a la trama un identificador de segmentación del mensaje y un indicador de finalización del mensaje, ya que es la mejor forma de llevar un orden y además de saber cuándo termina el atributo con registros múltiples.
- Para lograr el mejor resultado a lo largo del proyecto, se recomienda desarrollar los servicios en primer lugar a nivel del Servidor de procesos y luego importarlos en el Broker y generar la lógica correspondiente, ya que, si a nivel del Broker ya se tiene el archivo WSDL generado, se simplifica en gran medida la implementación a nivel del ESB.
- Luego de hacer el despliegue del proyecto a los servidores de la entidad bancaria, hacer las respectivas pruebas de estrés para determinar el comportamiento del sistema con diferentes cargas de usuario.
- Una vez desplegado el sistema de solicitud TDC, se recomienda expandir la optimización de los sistemas del banco, empezando con el desarrollo e implementación de un módulo automatizado de solicitud de créditos bancarios a través de internet, ya que ese módulo es muy similar al de Solicitud de Tarjetas de Crédito, por lo que se contaría con una gran experiencia para llevarlo a cabo de



forma exitosa, reduciendo el riesgo de errores a lo largo de las fase.

### REFERENCIAS BIBLIOGRAFICAS.

- [1] IBM. “About IBM” Disponible en Internet: <http://www.ibm.com/ibm/responsibility/basics.shtml>. Consultado el 2 de Agosto de 2011.
- [2] Endrei, Mark. “Patterns: Service-Oriented Architecture and Web Services” Disponible en Internet: <http://www.redbooks.ibm.com/redbooks/pdfs/sg246303.pdf>. Consultado el 8 de Septiembre de 2011.
- [3] Mendling, Jan “Business Process Execution Language for Web Service” Disponible en Internet: <http://www.mendling.com/publications/emisaforum06.pdf>.
- [4] Lakshmanan, Geetika. “Multi-module monitoring with the V7 WebSphere BPM suite, Part 2: Use new plug-ins for WebSphere Integration Developer V7 to achieve end-to-end business process monitoring”. Disponible en Internet: [http://www.ibm.com/developerworks/websphere/library/techarticles/1011\\_lakshmanan/1011\\_lakshmanan.html](http://www.ibm.com/developerworks/websphere/library/techarticles/1011_lakshmanan/1011_lakshmanan.html). Consultado el 10 de Febrero de 2012.
- [5] Filenet. The Synergy Between BPM & SOA. 2006. Disponible en: [http://www.financialtech-mag.com/\\_docum/109\\_Documento.pdf](http://www.financialtech-mag.com/_docum/109_Documento.pdf). Consultado el 25 de Febrero de 2012.
- [6] Keen, Martin. “Getting Started with WebSphere Enterprise Service Bus V6”. Junio, 2006. Disponible en Internet: <http://www.redbooks.ibm.com/redbooks/pdfs/sg247212.pdf>. Consultado el 20 de Septiembre de 2011.
- [7] IBM Haifa Research Lab. CBM- Component Business Modeling. Disponible en Internet: <http://www.research.ibm.com/haifa/projects/software/cbm/index.html>. Consultado el 24 de Febrero de 2012.
- [8] Goulikar, Balraj. “Speed CBS development using IBM WebSphere Business Fabric industry content packs.  
<http://www.ibm.com/developerworks/webservices/library/ws-cbsdev/>. Consultado el 24 de Febrero de 2012.

- [9] Ticknor, Margaret “IBM WebSphere Application Server V8 Concepts, Planning and Design Guide”. Disponible en Internet: <http://www.redbooks.ibm.com/redbooks/pdfs/sg247957.pdf>. Consultado el 8 de Septiembre de 2011.
- [10] IBM Information Center, “Documentación de IBM WebSphere Process Server” Disponible en internet: [http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp?topic=/com.ibm.wsps.602.doc/welcome\\_wps.html](http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp?topic=/com.ibm.wsps.602.doc/welcome_wps.html). Consultado el 12 de Agosto de 2011.
- [11] IBM, “SOW del proyecto SOA-BPM” Documento interno de IBM. Consultado el 12 de Agosto de 2011.
- [12] Davies, Saida “WebSphere Message Broker Basics” (Diciembre, 2005) Disponible en Internet: <http://www.redbooks.ibm.com/redbooks/pdfs/sg247137.pdf>.
- [13] Breakley, Darrell. “Connecting Your Business Using WebSphere Message Broker V7 as an ESB” (Mayo, 2010) Disponible en Internet: Consultado el 20 de Septiembre de 2011.
- [14] Tassilo, Andreas y Winter, Andreas “Model-Driven Software Migration” Disponible en Internet: <http://www.se.uni-oldenburg.de/documents/fuhr+2010.pdf>. Consultado el 27 de Diciembre de 2011.
- [15] IBM Method Web “Soma for AD” Documentos Internos de IBM. Consultado el 6 de Diciembre de 2011.
- [16] O'Reilly Technology School “What is a portlet” Disponible en Internet: <http://oreilly.com/java/archive/what-is-a-portlet.html> Consultado el 10 de Febrero de 2012.

## Apéndice A

### Tabla de Servicios Desarrollados.

**Tabla 1.** Servicios desarrollados durante el proyecto de pasantía.

Nombre de Servicio	Descripción
Cargar formulario de TDC	Dentro de él, se llaman todos los servicios que aportan datos al formulario en el cuál se hace la solicitud de TDC.
Consultar Agencias	Realiza la consulta de todas las agencias que se encuentran en el sistema legado.
Consultar Calle Avenida	Consulta las diferentes zonas de ubicación que puede seleccionar el usuario desde el formulario de solicitud de TDC.
Consultar Campo de Dirección	Consulta la información asociada a los Estados, Ciudades, Zonas Postales, Tipos de Residencias, Tipos de Dirección, Residencias, Códigos de Área y Calles Avenidas, disponibles en el sistema legado para cargar la dirección del Cliente.
Consultar Cargos Por Categoría Empresa	En este servicio se desarrolla el proceso de consulta de las Cargos asociados a las categorías de las empresas existentes.
Consultar cliente en el sistema legado X [uno para cada sistema legado]	Retorna la información de un cliente en caso de que exista, según el sistema legado que sea, se obtiene cierto tipo de información.
Consultar Código Destino del Requerimiento	Se consulta el código de destino del requerimiento, que será mostrado en el sistema para ser seleccionado por parte del Analista de Crédito.
Consultar Cuentas Bancarias	Se define el proceso de consulta de los tipos de cuentas bancarias que se definen basado en el número de cuentas que posee el solicitante
Consultar Edo Civil	Se define el proceso de consulta de la lista de Estados Civiles que serán mostradas en el sistema para ser seleccionado por parte del solicitante
Consultar Franquicias Banco	Se define el proceso de consulta de las franquicias ofrecidas por el banco para los productos TDC, serán mostradas en el sistema para que el Analista de Crédito la seleccione.
Consultar Lista Bancos	Consulta de la lista de Bancos que serán mostradas en el sistema para ser seleccionado por parte del solicitante. Por ejemplo en el caso de referencias bancarias.

Consultar Lista Ciudades	Consulta la lista de ciudades que serán mostradas en el formulario de solicitud de TDC.
Consultar Lista Códigos de Área	Consulta la lista de códigos de área que serán mostrados en el formulario de solicitud de TDC.
Consultar Lista de Cargos Automáticos	Define el proceso de consulta de las opciones de cargos automáticos que puede seleccionar el solicitante. Por ejemplo Cuenta Corriente Pago mínimo, Cuenta Corriente Deuda Total, etc.
Consultar Lista de Manejos Especiales	Consulta las condiciones de manejos especiales existentes, que serán mostradas en el sistema para que el Especialista de Negocios o Analista de crédito seleccione el tipo de manejo que se le dará a la solicitud.
Consultar Lista de Parentescos	Consulta la lista de parentescos que serán mostrados en el sistema para ser seleccionados por parte del solicitante.
Consultar Lista de Productos	Es utilizado para realizar la consulta al sistema legado de la lista con los distintos productos que se encuentran disponibles para su selección en el formulario de Solicitud de TDC.
Consultar Lista de Tipos de Dirección de Correspondencia	Permite realizar una consulta contra el sistema legado para obtener el listado de tipos de dirección para el envío de correspondencia, disponibles al momento de generar una nueva solicitud de TDC.
Consultar Lista Estados	Consulta la lista de estados que serán mostrados en el sistema para ser seleccionado por parte del solicitante.
Consultar Lista Profesiones	Este servicio es utilizado para realizar la consulta al sistema legado de la lista con las distintas profesiones que se encuentran disponibles para su selección en el formulario de Solicitud de TDC.
Consultar Lista Tipos Dirección de Envío de Plástico	se define el proceso de consulta de los posibles tipos de dirección de envío de plástico que puede seleccionar el solicitante. Por ejemplo Habitación, Oficina.
Consultar lista Zona Postal	Este servicio consulta la lista de zonas postales que pertenecen a una ciudad específica y que

	serán mostradas en el sistema para ser seleccionado por parte del solicitante.
Consultar Recaudos por Producto	Este servicio consulta los recaudos de un producto particular. Estos serán mostrados en el sistema para ser verificados por el especialista de Negocios al momento de recibir los recaudos del solicitante.
Consultar Solicitud Tarjeta de Crédito	El servicio comprende la consulta en Process Server de las solicitudes con instancias vigentes, de acuerdo a un conjunto de criterios de búsqueda específicos.
Consultar Tipo Mercado	Este servicio comprende la consulta de la lista de los tipos de mercado que serán mostradas en el sistema para ser seleccionado por parte del Especialista de negocios o Analista de crédito.
Consultar Tipos Cuentas	Comprende la consulta de los tipos de cuenta a ser cargados por el cliente en el formulario de solicitud de TDC.
Consultar Tipos de Residencia	Este servicio realiza la consulta de los tipos de residencia que puede escoger el solicitante, por ejemplo: Propia, Alquilada, De mis padres, etc.
Consultar Tipos Empresas	Define el proceso de consulta de los tipos de residencia que se le asignarán a los solicitantes basado en el lugar en donde vivan, estas serán mostradas en el sistema para ser seleccionado por parte del Analista de Crédito.
Consultar Zonas Vivienda	Este servicio consulta las zonas que se le asignan a una vivienda dependiendo de la locación donde se encuentre.
Crear cliente en el sistema legado.	Realiza la creación de un cliente nuevo en el sistema legado.
Crear Solicitud de Tarjeta de Crédito	Se encarga de crear a nivel de WPS una nueva instancia del Proceso Solicitud de Tarjeta de Crédito.
Generar Nombre Plástico	Este servicio comprende la generación del nombre que será impreso en el plástico de la tarjeta, según ciertas reglas establecidas.
Registrar Solicitud	Registra una solicitud en el sistema legado.

Consultar Tipos Solicitud de TDC	Este servicio es utilizado para realizar una consulta al sistema legado X. El legado retorna una lista con los distintos tipos de solicitudes que pueden ser seleccionados por los usuarios.
Consultar Residencias	El servicio comprende la consulta de los tipos de residencias posibles para que un usuario complete su dirección. Por ejemplo casa, quinta, edificio, galpón entre otros
Consultar Categoría Empresa	En este servicio se define el proceso de consulta de las categorías de las empresas existentes. El servicio las mostrará en el sistema para que el Analista de Crédito seleccione la Categoría de la empresa donde trabaja el solicitante.
Consultar Tipos de Dirección	Este servicio consulta la lista de tipos de dirección que serán mostradas en el sistema para ser seleccionado por parte del solicitante.

## Apéndice B

### Descripción de Subsistemas

**Tabla 2.** Descripción de los subsistemas presentes en el desarrollo del proyecto.

Nombre del subsistema	Descripción
<i>ComponenteSolicitudTDC</i>	Subsistema que provee la funcionalidad al usuario externo para ingresar la solicitud de TDC
<i>GUI</i>	Interfaces graficas para interacción con el sistema
<i>ActiveDirectory</i>	Autenticación y Autorización al sistema
<i>BRMS</i>	Sistema que administra y provee todas las reglas de negocio
<i>Solicitud de TDC</i>	Proceso de negocio para la solicitud de tarjetas de crédito
<i>MailServer</i>	Envío de mensajes vía correo electrónico
<i>Enterprise Service Bus</i>	Mediación y composición de servicios
<i>DispatcherPool</i>	Integración con los sistemas legado en ambiente iSeries
<i>ReporteSolicitud</i>	Genera un reporte imprimible de la solicitud de TDC realizada.

## **Apéndice C**

**Ejemplo de la información de un Servicio en el Portafolio de Servicios.**



<Proyecto SOA – BPM >

## Portafolio de Servicios

Versión <1.7>

¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.

## Historia de Revisiones

Fecha	Versión	Resumen de Cambios	Autor
11/08/2011	1.0	Inicio Redacción Documento	IBM
12/09/2011	1.1	Adición de nuevos servicios	IBM
14/09/2011	1.2	Refinar detalle del portafolio de servicios	IBM
15/09/2011	1.3	Cambios en la estructura del portafolio, incorporación de criterios y ajustes de peso	IBM
21/09/2011	1.4	Inclusión de observaciones, inclusión de nuevos servicios y reglas de negocio	IBM
22/09/2011	1.5	Incluir áreas Funcionales	IBM
04/10/2011	1.6	Inclusión de nuevos servicios	IBM
10/10/2011	1.7	Actualizaciones con casos de uso certificados	IBM

---

**Tabla de Contenido**

1.....	Introducción	
.....		64
2.....	Referencias	
.....		65
3.....	Portafolio de Servicios	
.....		66
3.1    Jerarquía de Servicios		66
3.2    Detalle del Portafolio de Servicios		67
3.2.1    Servicio: Cargar formulario de TDC		68
4.....	Abreviaturas	
.....		70

## 1. Introducción

El portafolio de servicios forma parte del modelo de servicios de la solución SOA/BPM de la entidad bancaria y consiste en la identificación y definición del conjunto de servicios candidatos que estarán soportando el proceso de negocio de solicitud de tarjeta de crédito.

El documento esta dividió en dos principales secciones.

Sección	Descripción	Identificación en SOMA
Portafolio de Servicios	Identificación de servicios derivados en la fase de descubrimiento de SOMA, organizado jerárquicamente mediante la descomposición funcional del proceso de negocio Solicitud de tarjeta de crédito	Identificación SOMA: <ul style="list-style-type: none"> <li>• Análisis de área de descomposición del dominio funcional</li> <li>• Dominio de Descomposición.</li> <li>• Metas del Modelado de Servicio</li> <li>• Análisis de archivos existentes</li> </ul>
Exposición de Servicios	Análisis del portafolio de servicios basado en diferentes criterios que determinan el grado de exposición que tendrá el servicio, el cual es determinado mediante la aplicación de un instrumento de medición (Service Litmus test). Los criterios son derivados y basados en las mejores prácticas de IBM y la industria en soluciones SOA	Especificación de servicios: <ul style="list-style-type: none"> <li>• Decisiones de exposición de servicios empleando el <i>Service Litmus Test</i> como parte de la racionalización y re-estructuración de servicios</li> </ul>

*Tabla 1: Detalle del Portafolio de Servicio*

---

## **2. Referencias**

Este documento se basa o hace referencia a los siguientes documentos:

1. Documento de reglas de negocio.
2. Documento de dependencias de servicios.

---

### **3. Portafolio de Servicios**

El Portafolio de Servicios, representa un listado de Servicios candidatos, los cuales debe ser clasificado y categorizados.

A continuación se define su jerarquía en función de la secuencia usada para implementación el proceso de negocio Solicitud de tarjeta de crédito y de su descomposición funcional.

---

#### **3.1 Jerarquía de Servicios**

##### **1. Cargar Formulario Solicitud TDC:**

###### **1.1 Crear solicitud (tipo de solicitud)**

###### **1.1.2 Consultar Campos Dirección**

###### **1.1.2.1 Consultar Lista Estado (Perfil de Cliente)**

###### **1.1.2.2 Consultar Lista Ciudades (Perfil de Cliente)**

###### **1.1.2.3 Consultar Lista Zona Postales (Perfil de Cliente)**

###### **1.1.2.4 Consultar Lista residencia**

###### **1.1.2.5 Consultar Lista código de áreas**

###### **1.1.2.6 Consultar Calle Avenida**

###### **1.1.2.7 Consultar Zonas Vivienda**

###### **1.1.3 Consultar Lista Profesiones**

###### **1.1.4 Consultar Lista Bancos (Hipotecas)**

###### **1.1.5 Consultar Agencias**

###### **1.1.6 Consultar Tipos Empresas**

###### **1.1.6.1 Consultar Cargos Por Categoría Empresa**

###### **1.1.7 Consultar Tipos Cuenta**

###### **1.1.8 Consultar Lista Productos (TDC)**

###### **1.1.8.1 Consultar Recaudos**

###### **1.1.9 Consultar Lista de Cargos automáticos**

###### **1.1.10 Consultar Características Contacto**

###### **1.1.11 Consultar Edo Civil**

###### **1.1.12 Consultar Tipo Cliente**

###### **1.1.13 Consultar Tipo Mercado**

###### **1.1.14 Consultar Lista de Parentescos**

###### **1.1.15 Consultar Tipos de Dirección Correspondencia**

###### **1.1.16 Consultar Tipos de Dirección de Envío de Plástico**

- 1.2 Consultar solicitud (TDC)
  - 1.2.2 Consultar Código Destino del Requerimiento
  - 1.2.3 Lista de Manejos Especiales
  - 1.2.4 Consultar Lista Estados Referencias Personales
- 1.3 Crear cliente en el sistema legado X
- 1.4 Insertar la Solicitud en el sistema legado X
- 1.5 Consultar Cliente en el sistema legado Y
- 1.6 Evaluar Solicitud
- 1.7 Actualizar Estado de la solicitud
- 1.8 Actualizar Datos Solicitud
- 1.9 Consultar información financiera
- 1.10 Consultar información financiera en el sistema legado X
- 1.11 Consultar información financiera en el sistema legado X
- 1.12 Consultar Buro Scoring cliente
- 1.13 Validar Bloqueo de Fondos (Solicitud TDC Credimaxima)
- 1.14 Consultar TDC Preaprobada
- 1.15 Consultar Ciclo de Facturación
- 1.16 Consultar Denominación TDC
- 1.17 Consultar Histórico de Solicitudes
- 1.18 Reasignación Tarea en sistema legado X.
- 1.19 Lista Edos Ref Personales
- 1.20 Consultar Lista Cargos Automáticos
- 1.21 Consultar Cuentas Bancarias
- 1.22 Generar Nombre Plastico

---

### 3.2 Detalle del Portafolio de Servicios

La identificación de servicios está dirigido a construir un portafolio de servicios que pueden estar asociados a las necesidades funcionales y metas del negocio, así como también sistemas existentes, siendo estos elementos los considerados para determinar que un servicio es candidato para ser expuesto.

Cada servicio se presentará de la siguiente manera:

**Servicio:** Nombre del Servicio

**Descripción:** Breve descripción del servicio o sus operaciones

**Propietario del Servicio:** Indica el ente que será propietario del servicio

**Procesos de Negocio:** Nombre del proceso de negocio al que el servicio estará asociado.

**Componentes de Negocio / Funciones del Negocio:** Nombre del componente o función del negocio a la que el servicio estará asociado.

**Meta:** Especificación de la metas o sub-metas a las que estará asociado el servicio.

**Activo Asociado al Servicio:** Descripción de la aplicación y/o legado que provee la funcionalidad expuesta por el servicio.

**Disposición del Servicio:** Se empleara la siguiente leyenda para identificar la disposición del servicio: "C" Servicio Candidato, "E" Servicio seleccionado para ser Expuesto, "NE" Servicio no seleccionado para ser Expuesto

### 3.2.1 Servicio: Cargar formulario de TDC

**Descripción:** Servicio creado para proveer el formulario requerido donde el cliente deberá colocar los datos necesarios para crear una solicitud de tarjeta de crédito

**Propietario del Servicio:** Entidad bancaria

**Proceso de Negocio:** Solicitud de TDC

**Componente de Negocio / Funcionalidad del Negocio:** Cliente, Agencia

**Activo Asociado al Servicio:** Websphere Application Server el cual provee una aplicación J2EE

**Disposición del Servicio:** NE

---

### 3.3 Factorización y Racionalización de Servicios

Con el objetivo de optimizar el portafolio de servicio, cada servicio es analizado y en función de un criterio se identifica cuales servicios deben ser expuestos.

Los criterios basados en las mejores prácticas de la industria se describen a continuación:

**Alineación al negocio:** Evalúa el nivel de correspondencia entre el negocio y el servicio, basado en los siguientes puntos:

Funcionalidad empresarial de soporte a los procesos y objetivos empresariales

Intención de financiar el servicio en todo su ciclo de vida

Uso y exposición del servicio hacia socios externos y/o potenciales clientes.

Existencia actual y en el futuro de reutilización del servicio

**Servicio Componible:** Determina la posibilidad que el servicio participe en la composición de otros servicios los cuales son determinados por los siguientes puntos:



El servicio soporta los requerimientos no funcionales

El servicio es sin estado

El servicio es independiente y puede cumplir por si solo un objetivo empresarial

El servicio puede cooperar con otros servicios para cumplir objetivos empresariales

La tecnología de implementación del servicio es neutral

**Descripción:** Evalúa las capacidades del servicio de poder auto describirse, basados en los siguientes puntos:

El servicio posee una descripción externa la cual es distinta e independiente de la implementación física subyacente

El servicio puede ser descubierto y enlazado con la descripción del mismo

Contiene la descripción del servicio metadatos que permite la identificación del mismo.

**Reusabilidad:** Evalúa las capacidades del que el servicio pueda ser re usado por diferentes procesos de negocio

**Viabilidad Técnica:** La viabilidad técnica asegura que el servicio se pueda implementar y desplegar, basado en los siguientes puntos:

El esfuerzo de implementación y gestión del servicio es razonable y fácilmente viable

El servicio puede ser implementado por un número mínimo de sistema proveedores

No existen retos tecnológicos explícitos conocidos ni anticipados

A continuación se muestra una tabla donde basado en los siguientes criterios se clasifican y selecciona cuales servicios serán expuestos:

<b>Nombre del Servicio</b>	<b>Alineación al negocio</b>	<b>Servicio Componible</b>	<b>Descripción</b>	<b>Reusabilidad</b>	<b>Viabilidad Técnica</b>	<b>Decisión</b>
Cargar Formulario Solicitud TDC	No	No	Si	No	No	No

#### 4. Abreviaturas

<b>Abreviatura</b>	<b>Significado</b>
WPS	Websphere Process Server
TDC	Tarjeta de Crédito
J2EE	Plataforma empresarial de Java, correspondiente a sus siglas en ingles “Java Platform, Enterprise Edition”
SOA	Arquitectura Orientada a Servicios

## **Apéndice D**

### **Modelo de Especificación de Servicios**

**Proyecto SOA – BPM**

## **Especificación del Servicio**

**Consultar Agencias**

**Versión 1.4**

**¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

## Historia de Revisiones

Fecha	Versión	Resumen de Cambios	Autor
18/11/2011	1.0	Creación del documento	IBM.
21/11/2011	1.1	Modificación de los componentes.	IBM.
24/11/2011	1.2	Adaptación al documento con la plantilla generada.	IBM.
25/11/2011	1.3	Corrección del documento según observaciones recibidas.	IBM.
01/12/2011	1.4	Actualización de los nombres de los execution groups y las colas.	IBM.

---

## Tabla de Contenido

1. ....	Introducción	
.....		75
2. ....	Referencias	
.....		77
3. ....	Especificación del Servicio	
.....		78
3.1	Atributos relacionados con el contexto	78
3.2	Operaciones del Servicio	78
3.3	Eventos	79
3.4	Reglas y políticas	79
3.5	Diagrama de Contexto de Servicios	79
3.6	Mensajes de Servicios	80
3.6.1	Diagrama de Clases Mensaje de Entrada	80
3.6.2	Diagrama de Clases Mensaje de Salida	80
3.6.3	Mensajes de error	82
3.7	Entradas y Salidas del Programa RPG	83
3.7.1	Entrada del Programa RPG <nombre programa>	83
3.7.2	Salida del Programa RPG	83
3.7.3	Consideraciones para la salida del Programa RPG <nombre programa>	84
3.8	Requerimientos No funcionales de Servicios	84
3.9	Requerimientos Funcionales de Servicio	86
3.10	Correspondencia Servicio – Componentes	86
3.11	Decisiones de Realización	87
3.11.1	Decisiones de Arquitectura para la Realización del servicio	87

## 1. Introducción

La especificación de servicio es parte del Modelo de Servicios, se emplea para ubicar un servicio en contexto, describiendo sus atributos, operaciones, eventos y sus respectivas correspondencias con los componentes, reglas y políticas, para cada uno de los servicios a desarrollar.

Este documento debe manejarse en conjunto con el Portafolio de Servicios y el Modelo de Servicios.

Las secciones contempladas en este documento se especifican en la siguiente tabla:

Nro Sección	Sección	Descripción
3.1	Atributos relacionados con el contexto	Especifica los atributos relacionados con el contexto para el ensamblaje dinámico (ensamblar múltiples mensajes en un flujo) y enrutamiento de mensajes (mensajes que cambian de ruta en un flujo dinámicamente en base a este atributo de contexto)
3.2	Operaciones del Servicio	Identifica las Operaciones del Servicio
3.3	Eventos	Identifica los eventos que dan inicio a los servicios y sus operaciones

3.4	Reglas y políticas	Especificación/Asociación de las reglas ejecutadas por la operación del servicio y las políticas que sigue el mismo.
3.5	Diagrama de contexto del servicio	Ilustra los consumidores y proveedores del servicio
3.6	Mensajes del Servicio	Mensajes que se intercambian entre los consumidores del servicios y el proveedor del servicio
3.7	Entradas y Salidas del programa RPG	Definición de las tramas de entrada y salida para los programas RPG que responden a las solicitudes del servicio.
3.8	Requerimientos No funcionales del Servicio	Requerimientos no funcionales que determinan las características del servicio
3.9	Requerimientos Funcionales del Servicio	Requerimientos funcionales que determinan las características del servicio
3.10	Correspondencia Servicios - Componentes	Mapeo de los componentes de servicio con los Componentes Técnicos y Funcionales
3.11	Decisiones de Arquitectura para la realización de un servicio	Expone las decisiones arquitectónicas que sustentan la razón de la realización de un servicio.

**Tabla 1: Elementos de la Especificación de Servicios**



---

## **2. Referencias**

Este documento se basa o hace referencia a los siguientes documentos:

1. *Especificación Requerimientos Solución v2.0.doc*
2. *Decisiones Arquitectónicas v1.5.doc*
3. *Modelo de Componentes v1.1.doc*
4. *CU\_GF\_UTIL\_09\_Consultar\_Agencias v1.0.doc*

---

### 3. Especificación del Servicio

En este servicio se define todo el proceso de consulta de Agencias en el legacy X.

---

#### 3.1 Atributos relacionados con el contexto

Esta sección del documento describe los atributos en el contexto de la ejecución de un servicio que pueden afectar el flujo de los mensajes del servicio o el ensamblado de múltiples mensajes.

Atributo	Descripción	Tipo de Dato	Posibles Valores
indicadorInicioMensaje	Atributo que indica si la trama contiene el primer mensaje.	Character	1 – Indica que es el primer mensaje  0 – Indica que no es el primer mensaje
indicadorFinalMensaje	Mensaje que contiene un único valor que indica que ya no existen mensajes	Character	“fin”

Tabla 2: Atributos relacionados con el contexto

---

#### 3.2 Operaciones del Servicio

Esta sección del documento describe las diferentes operaciones que contiene el servicio para exponer su funcionalidad.

##### OP01: consultarAgencias

- La operación “*consultarAgencias*” realiza la consulta de todas las agencias que se encuentra en el legado X.

---

### 3.3 Eventos

Esta sección del documento describe los eventos del proceso de negocio que desencadenan las operaciones del servicio.

#	Operación del servicio	Evento	Reglas y políticas
1	OP01	El solicitante debe ingresar al portal web y acceder al formulario de solicitud de productos.	NA

**Tabla 3: Operaciones del servicio, eventos, reglas y políticas**

---

### 3.4 Reglas y políticas

Este servicio describe reglas y políticas generales que aplican al desarrollo de este servicio.

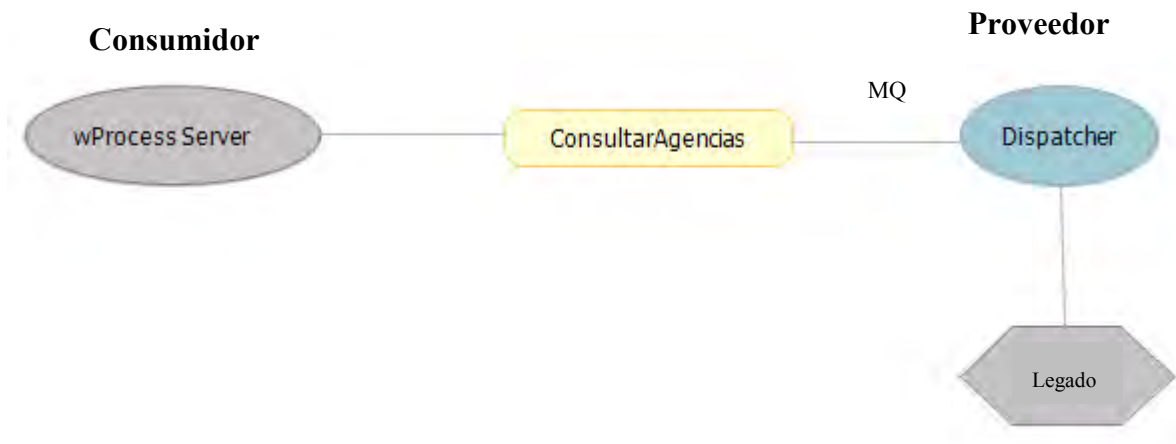
3.4.1. Este servicio puede ser invocado por cualquier otro sistema.

3.4.2. Requiere un token de seguridad valido para la autenticación correspondiente.

---

### 3.5 Diagrama de Contexto de Servicios

Esta sección del documento describe el contexto de todas las entidades que interactúan con el servicio descrito en este documento.

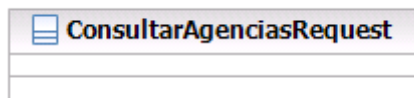


### 3.6 Mensajes de Servicios

Esta sección del documento describe los mensajes de entrada y salida para las diferentes operaciones del servicio, representadas como diagramas de clases.

#### 3.6.1 Diagrama de Clases Mensaje de Entrada

##### 3.6.1.1 OP01: ConsultarAgencias



#### 3.6.2 Diagrama de Clases Mensaje de Salida

##### 3.6.2.1 OP01: ConsultarAgencias



### 3.6.2.2 Clase: ConsultarAgencias

#	Nombre	Descripción	Formato	Obligatorio (si/no)
1	listaAgencias	Lista de objetos del tipo Entidad que contienen las agencias disponibles en el legado	N/A	Si

**Tabla 3** Clase *ConsultarAgenciasResponse*

### 3.6.2.3 Clase: Entidad

#	Nombre	Descripción	Formato	Valor	Obligatorio (si/no)
1	codigo	Identificador asociado a la agencia	NA	NA	Si

2	nombre	Nombre de la agencia	NA	NA	Si.
---	--------	-------------------------	----	----	-----

**Tabla 5: Clase Entidad**

### 3.6.3 Mensajes del Servicio

*MI-XXX* Mensaje Informativo

*ME-XXX* Mensaje de Error

*MA-XXX* Mensaje de Advertencia

Código	Mensaje Administrativo	Mensaje Funcional
MI0007	N/A	No se encuentran agencias registradas
ME0004	Servicio: [Nombre del servicio]. Plataforma Legado X no disponible	Ha ocurrido un error durante la consulta de datos.
ME0007	Servicio:[Nombre Servicio]. Cola [Nombre de la cola] no disponible.	Ha ocurrido un error. Por favor intente más tarde.

**Tabla 6: Mensajes del Servicio**

### 3.7 Entradas y Salidas del Programa RPG

Esta sección del documento define las tramas de entrada y salida para los programas RPG que van a responder a las solicitudes realizadas por este servicio.

Las tablas en esta sección del documento, representan tramas que serán interpretadas por programas RPG y serán generadas con el siguiente formato:

Orden:	1	2	3	4	5	6		n
	IdSesion	IdUsuario	idPrograma	version	cantidadTramasResponse	campo 1	...	campo n
Longitud:	30	15	10	4	4	n		n

#### 3.7.1 Entrada del Programa RPG <nombre programa>

Orden	Body	Campo	Tipo de Dato	Longitud
1	Metadata	idSesion	Char	30
2		idUsuario	Char	15
3		idPrograma	Char	10
4		version	Char	4
5		cantidadTramasResponse	Char	4

Tabla 7: Entradas del programa RPG.

#### 3.7.2 Salida del Programa RPG

Orden	Body	Campo	Nombre Campo en el 400	Tipo de Dato	Longitud
1	Metadata	numeroMensaje	N/A	Char	1
2	Contenido	codigo	OFCDOF	Char	4
3		nombre	OFOFCN	Char	30

**Tabla 8: Salidas del programa RPG**

### 3.7.3 Consideraciones para la salida del Programa RPG <nombre programa>

Criterio	Valor
Origen(Legado)	Legado X
Tabla Origen	TABLAORIGEN
Cantidad Registros que existen en el legado	110
¿Se puede manejar en una sola ráfaga? (Si/No)	Si
Cantidad de Ráfagas	1

**Tabla 9: Consideraciones para la salida del RPG.**

## 3.8 Requerimientos No funcionales de Servicios

Esta sección describe los requerimientos no funcionales que tienen impacto en las decisiones de realización del servicio que describe este documento.

#	Operación del Servicio	Requerimientos No funcionales (RNF)	
		Tipo RNF	RNF
1	OP01	Capacidad y Desempeño	Requerimiento Desempeño VOL001 <Tiempo de respuesta>
2	OP01	Capacidad y Desempeño	Requerimiento Desempeño



			VOL003 <Número total de usuarios >
3	OP01	Capacidad y Desempeño	Requerimiento Desempeño VOL004 <Número máximo de usuarios concurrentes>
4	OP01	Seguridad	Requerimiento Seguridad SEG007 <Autenticación>
5	OP01	Estándares Técnicos	Requerimiento Restricción EST001 <Estándares JAVA>
6	OP01	Estándares Técnicos	Requerimiento Restricción EST002 <Nomenclatura de Servicios Web>
7	OP01	Estándares Técnicos	Requerimiento Restricción EST005 <Trazas de auditoría>

**Tabla 10: Requerimientos No Funcionales del Servicio**

---

**3.9 Requerimientos Funcionales de Servicio**

Esta sección del documento describe los requerimientos funcionales que describen la funcionalidad de la solución que deben ser implementadas en el servicio que describe este documento.

Los requerimientos funcionales para el servicio “Consultar Agencias” se encuentran definidos en el documento de Caso de Uso CU\_GF\_UTIL\_09\_Consultar\_Agencias v1.0.doc

---

**3.10 Correspondencia Servicio – Componentes**

Esta sección del documento mapea cuales componentes en la arquitectura de la aplicación deben contener la implementación del servicio que describe este documento.

#	Operación del servicio	Correspondencia de servicios a los componentes		
		Componente de servicio	Componente funcional	Componente técnico
1	OP01	ComponenteSolicitudTDC, Dispatcher	FormularioTDC.	GestorMQ, ColaRequest, ColaResponse, ProgramaRPG, Listener, FlujoMediacion

**Tabla 11: Correspondencia de servicios**

### 3.11 Decisiones de Realización

#### 3.11.1 Decisiones de Arquitectura para la Realización del servicio

**Tabla 12: Resumen de decisiones de arquitectura para la realización del servicio**

Punto de decisión	Decisión
Enrutamiento a aplicaciones legadas	DA0005: Envío de la metadata para el dispatcher a través del encabezado del mensaje MQ, para que este pueda realizar sus tareas de enrutamiento de mensajes.
Integración con los legados	DA0001: Uso de colas MQ para el llamado de programas RPG.
Registro del servicio	El servicio se encontrara registrado en Websphere Registry and Repository.
Exposición del Servicio	Web Services expuesto por un flujo de mediación en Websphere Message Broker
Formato del mensaje	El flujo de mediación convertirá un mensaje en formato XML en una trama para ser consumida por los legados. En dirección contraria, se recibirá una trama que será modificada en XML dentro del flujo de mediación.
Despliegue del Servicio	Este servicio será desplegado en el execution group FAC0001 para la instancia de broker VE002
Colas de response y request para el servicio	Cola Request: FAC0001.REMOTE.REQUEST ColaResponse: FAC0001.LOCAL.RESPONSE

## **Apéndice E**

### **Modelo de Componentes.**

**Proyecto SOA – BPM**

## **Modelo de Componentes**

**Versión 1.4**

## Historia de Revisiones

Fecha	Versión	Resumen de Cambios	Autor
29/08/2011	1.0	Creación del documento.	IBM

---

## Tabla de Contenido

1.....	Introducción	
.....		93
1.1	Identificación	93
1.2	Referencias	93
1.3	Definiciones	93
2.....	Modelo de Componentes para SOMA	
.....		94
2.1	Vista general del Sistema y relaciones	94
2.2	Descripciones de Subsistemas	96
2.2.1	Subsistema ComponenteSolicitudTDC	96
2.2.2	Subsistema GUI	99
2.2.3	Subsistema ActiveDirectory	103
2.2.4	Subsistema BRMS	106
2.2.5	Subsistema Solicitud de TDC	109
2.2.6	Subsistema Enterprise Service Bus	114
2.2.7	Subsistema DispatcherPool	117
2.3	Atributos de Componentes	119
2.3.1	Componentes para Subsistema ComponenteSolicitudTDC	119
2.3.2	Componentes para Subsistema GUI	123
2.3.3	Componentes para Subsistema ActiveDirectory	125
2.3.4	Componentes para Subsistema BRMS	128
2.3.5	Componentes para Subsistema Solicitud de TDC	131
2.3.6	Componentes para Subsistema Enterprise Service Bus	136

2.3.7	Componentes para Subsistema <DispatcherPool>	139
2.4	Correspondencia de Componentes con las capas de Arquitectura	<b>¡Error! Marcador no definido.</b>
2.5	Abreviaturas	142



---

## 4. Introducción

El análisis de subsistemas, describe a los mismos en términos de responsabilidades, composición, comportamiento interfaces y relación entre todos los subsistemas. La composición interna de los subsistemas se describe identificando a los componentes de servicios, componentes técnicos que cada subsistema va a contener. Estos elementos serán descritos mediante una combinación de diagramas de UML y descripciones textuales.

Los componentes de servicio son asociados con los subsistemas que están funcionalmente alineados con el negocio y las áreas funcionales. La relación provee trazabilidad y justificación para su inclusión en la solución, un concepto clave en la metodología SOMA.

Un componente de servicio es la realización de un subsistema, una agrupación lógica de servicios alineados funcionalmente, que tienen suficiente importancia para el negocio. El modelo de componentes documenta los componentes de servicio, componentes funcionales, componentes técnicos y subsistemas que realizan los aspectos funcionales de una solución basada en SOA.

---

### 4.1 Identificación

Este documento describe el modelo de componentes SOMA, para el proceso “Solicitud de Tarjeta de Crédito” para el proyecto SOA-BPM de la entidad financiera.

---

### 4.2 Referencias

Este documento se basa o hace referencia a los siguientes documentos:

N/A

---

### 4.3 Definiciones

**Widget:** Un widget es un componente Web que contiene funcionalidad propia y puede ser embebido en una página Web para proveer contenido dinámico. En el contexto de la solución SOA/BPM, los widgets serán componentes independientes que pueden incorporarse al portal de negocio (Business Space) para proveer de las funcionalidades requeridas al usuario final. A diferencia de otras aplicaciones web, el widget es un componente reutilizable que se puede incorporar en el espacio de negocio de cualquier usuario que tenga permisos para utilizarlo.

---

## 5. Modelo de Componentes para SOMA

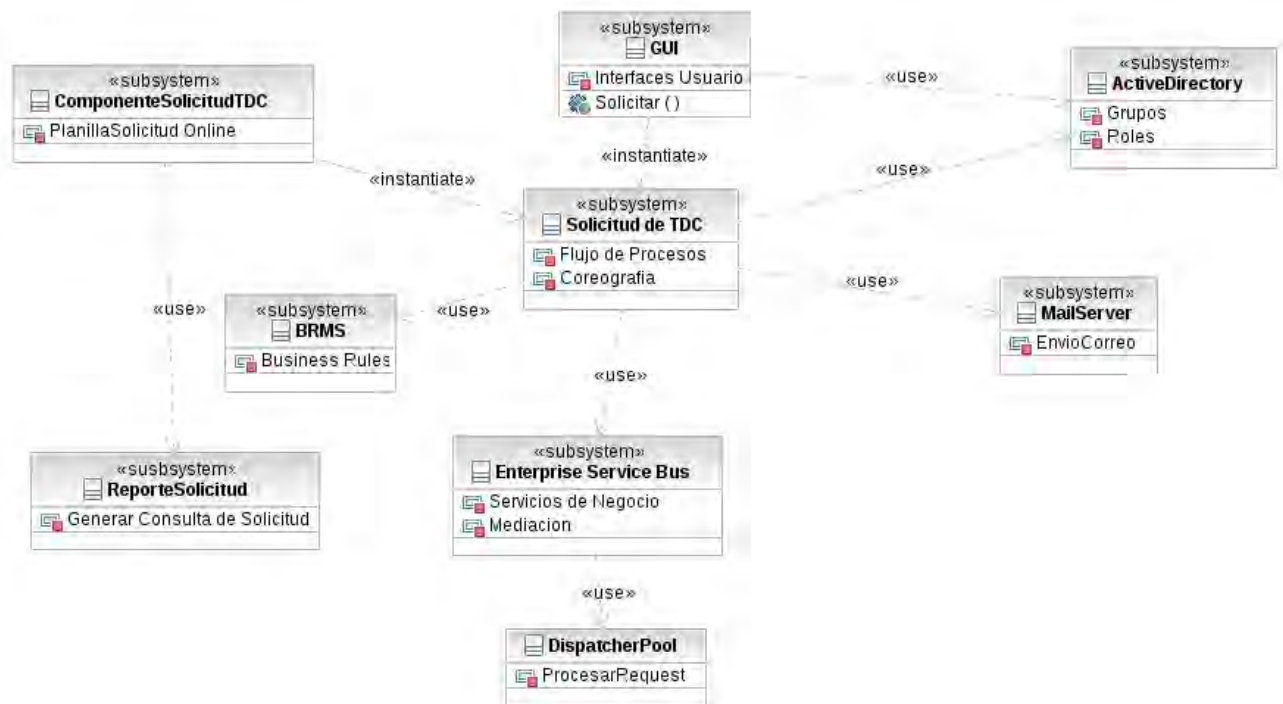
Tanto el análisis de subsistemas como la especificación de componentes utilizan primariamente diagramas de clase y de interacción de UML, junto a descripciones textuales para representar los siguientes aspectos de una solución basada en SOA:

- Vista general de subsistemas y relaciones
- Descripción de subsistemas, composiciones, interfaces y comportamiento
- Atributos y mensajes relacionados al componente de servicio, componentes funcionales y técnicos
- Interfaces e interacciones de componentes
- Asignación de componentes a capas de arquitectura

---

### 5.1 Vista general del Sistema y relaciones

Esta sección provee una vista general de las relaciones entre los subsistemas que se encuentran dentro del alcance del análisis. Un diagrama de relaciones y la tabla que lista los subsistemas son utilizadas para describir los subsistemas envueltos en la solución. Esta sección se genera típicamente de forma incremental, es revisada y extendida a medida que cada subsistema es analizado y documentado.



**Figura 5.1: Relación de subsistemas**

La Figura 1, describe las relaciones entre los diferentes subsistemas involucrados en la solución y sus responsabilidades dentro de la misma. A continuación, en la Tabla 1 se describe brevemente cada uno de estos subsistemas:

Tabla 1: Listado de subsistemas

Nombre del subsistema	Descripción
<i>ComponenteSolicitudTDC</i>	<i>Subsistema que va a proveer la funcionalidad al usuario externo para ingresar la solicitud de TDC</i>
<i>GUI</i>	<i>Interfaces graficas para interacción con el sistema</i>
<i>ActiveDirectory</i>	<i>Autenticación y Autorización al sistema</i>
<i>BRMS</i>	<i>Sistema que administra y provee todas las reglas de negocio</i>
<i>Solicitud de TDC</i>	<i>Proceso de negocio para la solicitud de tarjetas de crédito</i>
<i>Enterprise Service Bus</i>	<i>Mediación y composición de servicios</i>
<i>DispatcherPool</i>	<i>Integración con los sistemas legado en ambiente iSeries</i>
<i>ReporteSolicitud</i>	<i>Genera un reporte imprimible de la solicitud de TDC realizada.</i>

---

## 5.2 Descripciones de Subsistemas

Esta sección utiliza una combinación de descripciones textuales y diagramas UML para capturar la descripción de cada subsistema que componen la solución SOA/BPM.

### 5.2.1 Subsistema **ComponenteSolicitudTDC**

#### 5.2.1.1 Descripción del Subsistema.

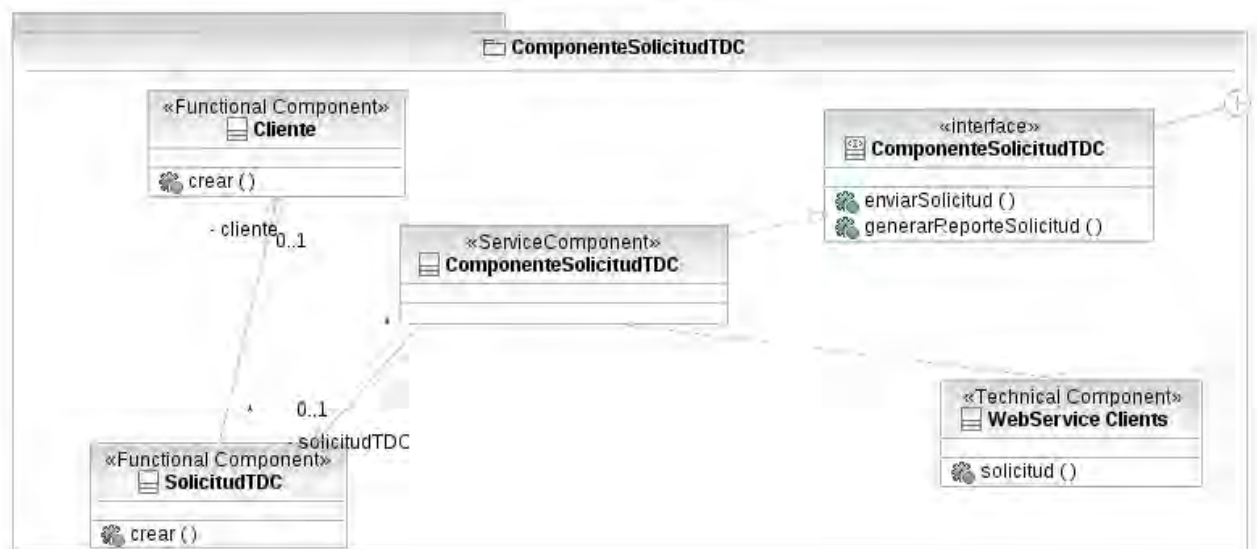
Esta sección describe las responsabilidades funcionales del subsistema ComponenteSolicitudTDC.

**Tabla 2: Descripción del Subsistema ComponenteSolicitudTDC.**

<b>Función</b>	<b>Descripción</b>
<i>Desplegar el formulario de solicitud de tarjeta de crédito de TDC</i>	<i>Desplegar el formulario de solicitud de tarjeta de crédito para los usuarios externos que deseen hacer una solicitud.</i>

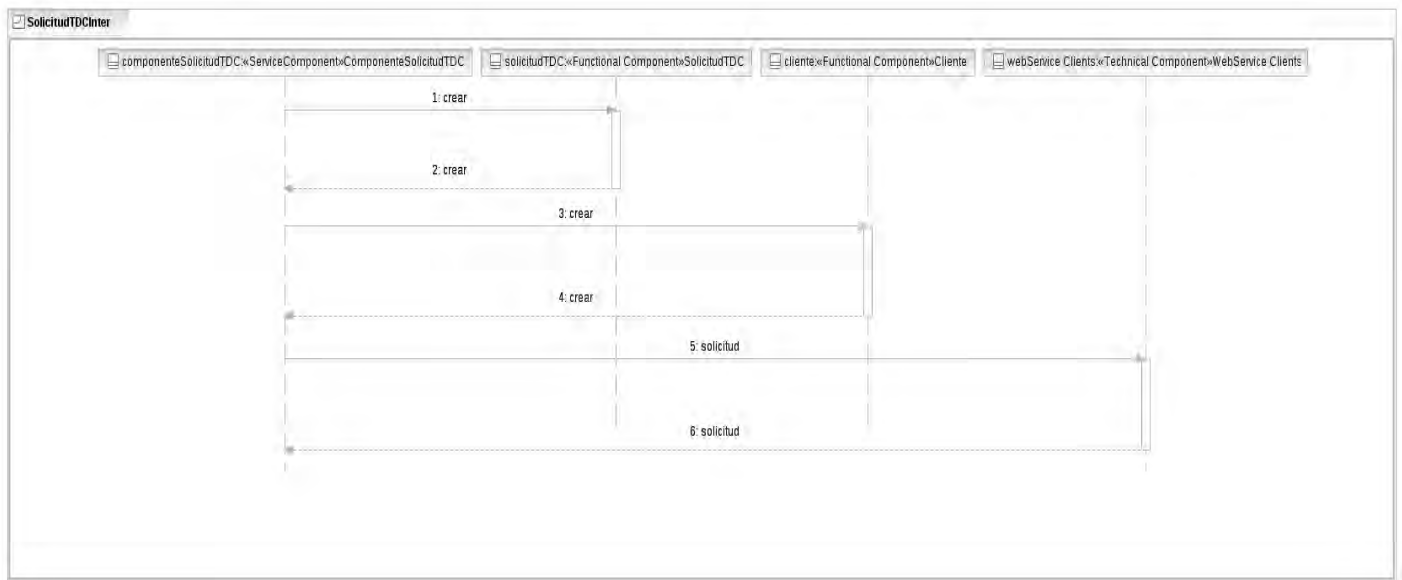
### 5.2.1.2 Composición e interfaces del Subsistema

La Figura 2 representa el diagrama de clases mostrando los componentes funcionales, técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El subsistema ComponenteSolicitudTDC, es responsable de proveer el formulario web a través del cual los usuarios externos enviarán la solicitud al sistema, para generar una nueva instancia del proceso de negocio “Solicitud de Tarjeta de Crédito”. Esta nueva instancia se genera a través de un WS expuesto por el componente “c” implementado en WebSphere Process Server para enviar los datos de la solicitud al proceso de negocio.



**Figura 5.2: Composición del subsistema ComponenteSolicitudTDC**

La Figura 3 representa un diagrama de secuencia que muestra una vista general de la interacción de los componentes que conforman el subsistema “ComponenteSolicitudTDC”.



**Figura 5.3: Interacción de componentes**

### 5.2.1.3 Comportamiento del Subsistema

El subsistema “ComponenteSolicitudTDC” es una aplicación J2EE desplegada en un WebSphere Application Server que genera el formulario Web para el ingreso de una nueva solicitud de tarjeta de crédito para usuarios externos. El formulario contiene todos los atributos requeridos para el ingreso de una solicitud en el Banco.

Se accede al formulario con un navegador estándar tal como Internet Explorer o Mozilla Firefox vía https para usuarios externos (clientes o no clientes).

## 5.2.2 Subsistema GUI

### 5.2.2.1 Descripción del Subsistema.

Esta sección describe las responsabilidades funcionales del subsistema GUI.

Función	Descripción
<i>Desplegar los widgets para interacción con el usuario</i>	<i>Desplegar los widgets</i>

**Tabla 3: Descripción del Subsistema GUI**

### 5.2.2.2 Composición e interfaces del Subsistema

La Figura 4 representa el diagrama de clases mostrando los componentes técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El subsistema GUI, es responsable de proveer las interfaces gráficas para que los usuarios internos del Banco puedan interactuar con el proceso de negocio “Solicitud de Tarjeta de Crédito”.

Esta funcionalidad es implementada a través de la herramienta Business Space. Business Space es un portal web que va a contener los “widgets” requeridos para la interacción con el mencionado proceso. El task widget permitirá ubicar el proceso deseado de negocio, de todas las instancias disponibles en un momento dado y los Human Task Widgets proveerán los formularios y componentes gráficos para que los usuarios ingresen data y actualicen el estado del proceso de negocio.

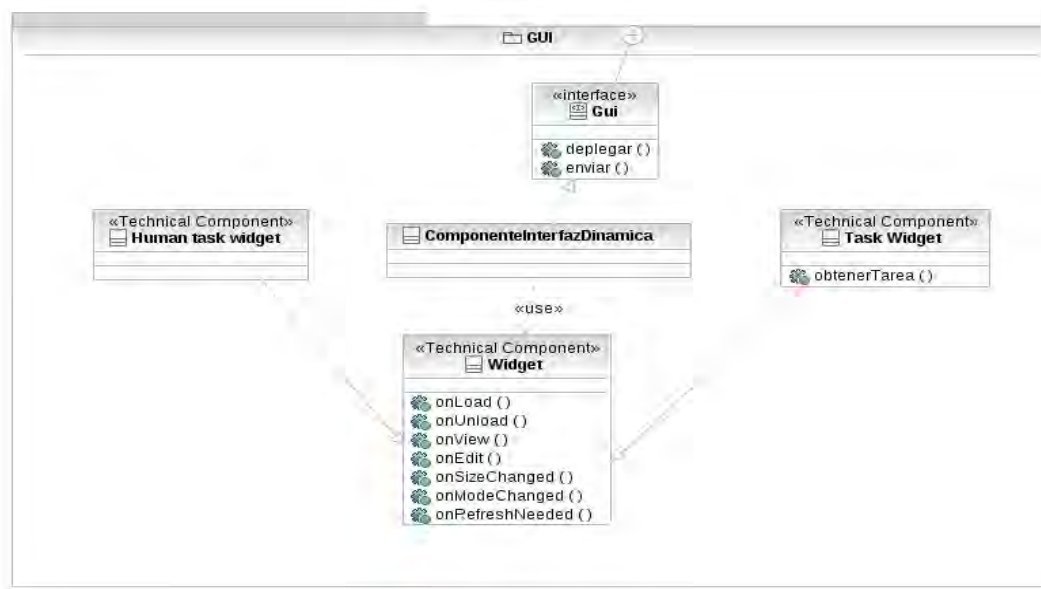
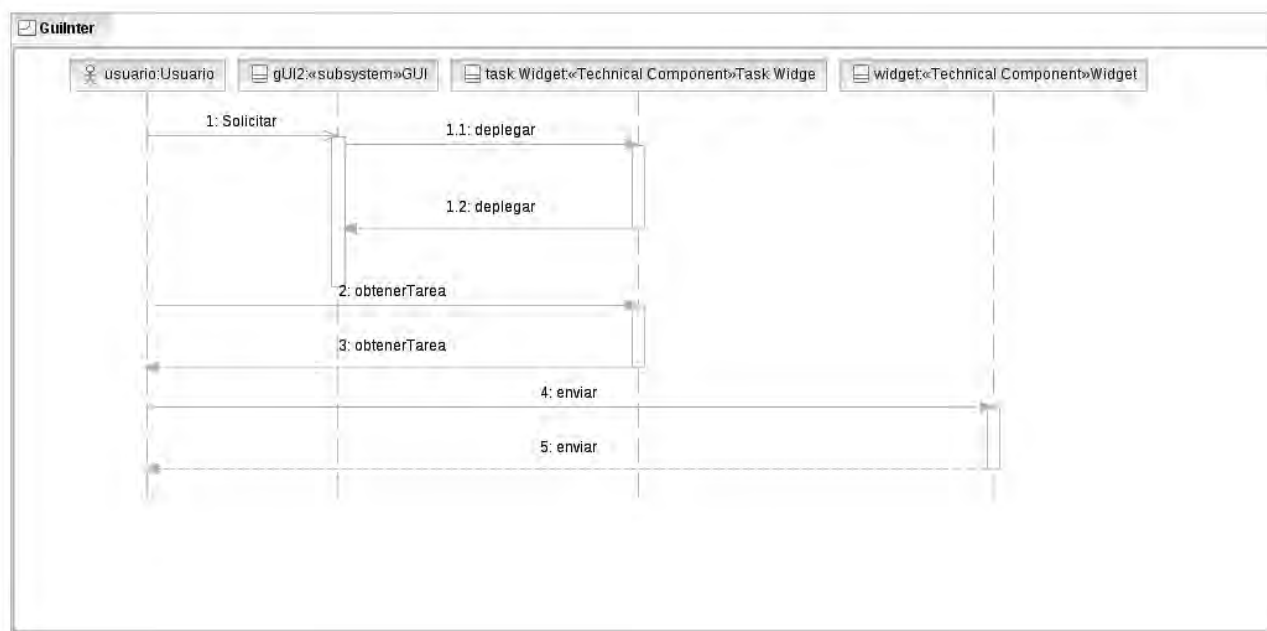


Figura 5.4: Composición del subsistema GUI



La Figura 5 representa un diagrama de secuencia que muestra una vista general de la interacción de los componentes que conforman el subsistema “GUI”.



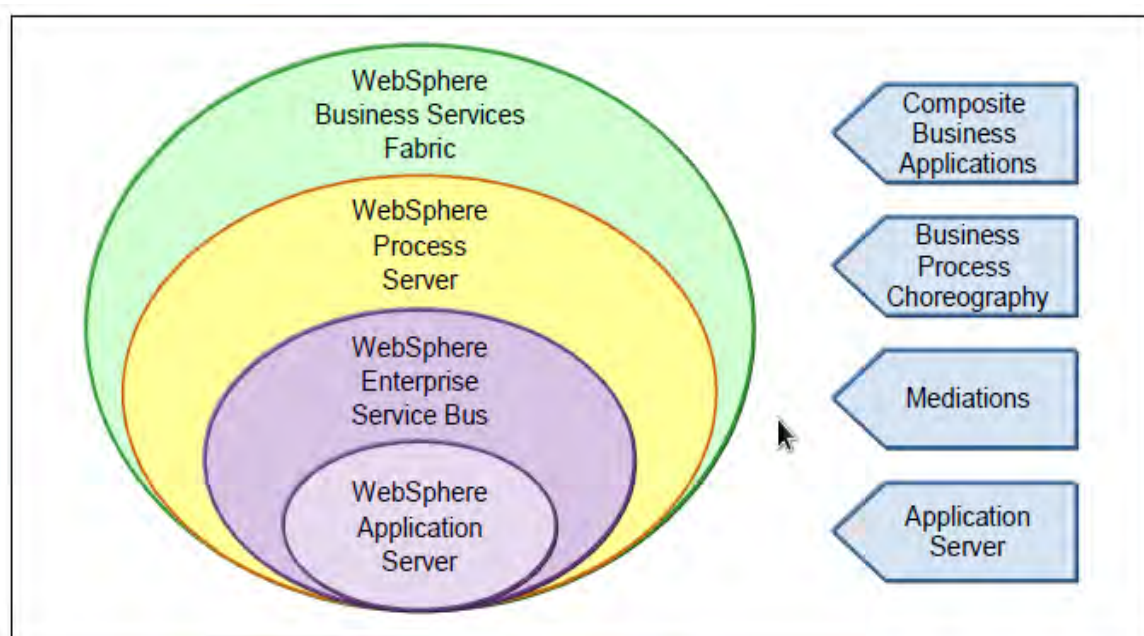
**Figura 5.5: Interacción de componentes**

### 5.2.2.3 Comportamiento del Subsistema

El subsistema “GUI” es soportado principalmente por el producto IBM Business Space powered by WebSphere. Business Space es un Framework para generación de interfaces gráficas para ser desplegadas por un navegador de Internet tal como Internet Explorer o Mozilla Firefox.

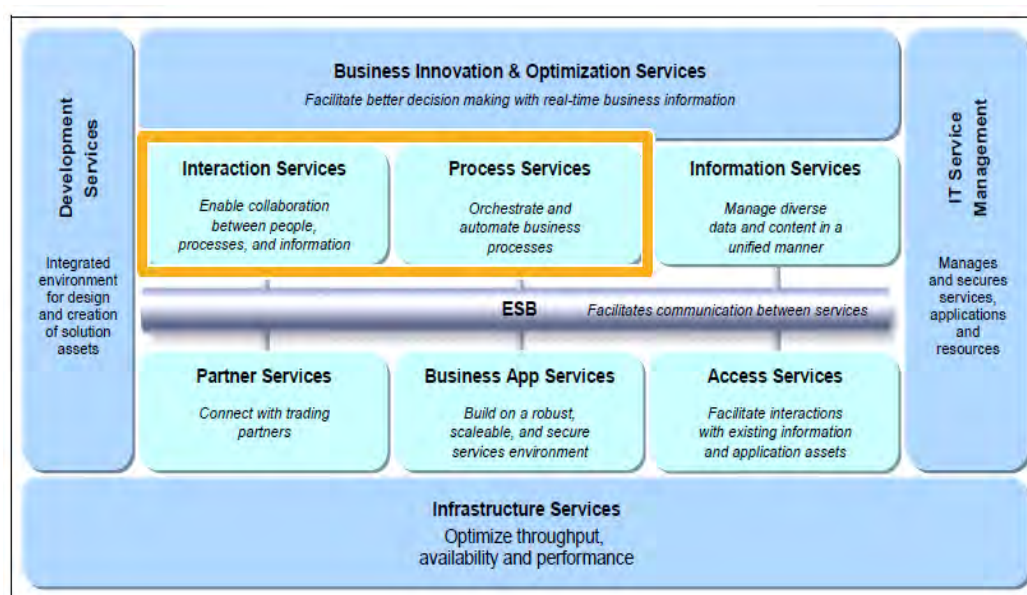
Con el uso de WebSphere Business Space, manejaremos el concepto de “*Business Space*”, que se refiere a una colección de contenido web especializado para el manejo del negocio. En el contexto de la solución SOA/BPM el Business space contendrá “widgets” para el manejo del proceso de negocio “Solicitud de Tarjeta de crédito”

Los widgets desarrollados para un nuevo Business Space deben ser desplegados con un EAR en un servidor. El runtime del Websphere Business Space es soportado por un IBM Websphere Application Server en su nivel mas básico, sin embargo también requiere del runtime de un IBM Websphere Process Server para poder soportar la funcionalidad de un proceso de negocio. A continuación en la figura 6 se puede observar la jerarquía de los runtimes soportados por IBM Websphere Business Space.



**Figura 5.6: Jerarquía de runtimes compatibles con Business Space**

En la arquitectura referencial de Websphere para soluciones de integración, Business Space provee las interfaces de usuario, en la figura 7 podemos ver el papel de Websphere Business Space en esta arquitectura.



**Figura 5.7: Business Space en la arquitectura referencial de integración de Websphere**

### 5.2.3 Subsistema **ActiveDirectory**

#### 5.2.3.1 Descripción del Subsistema.

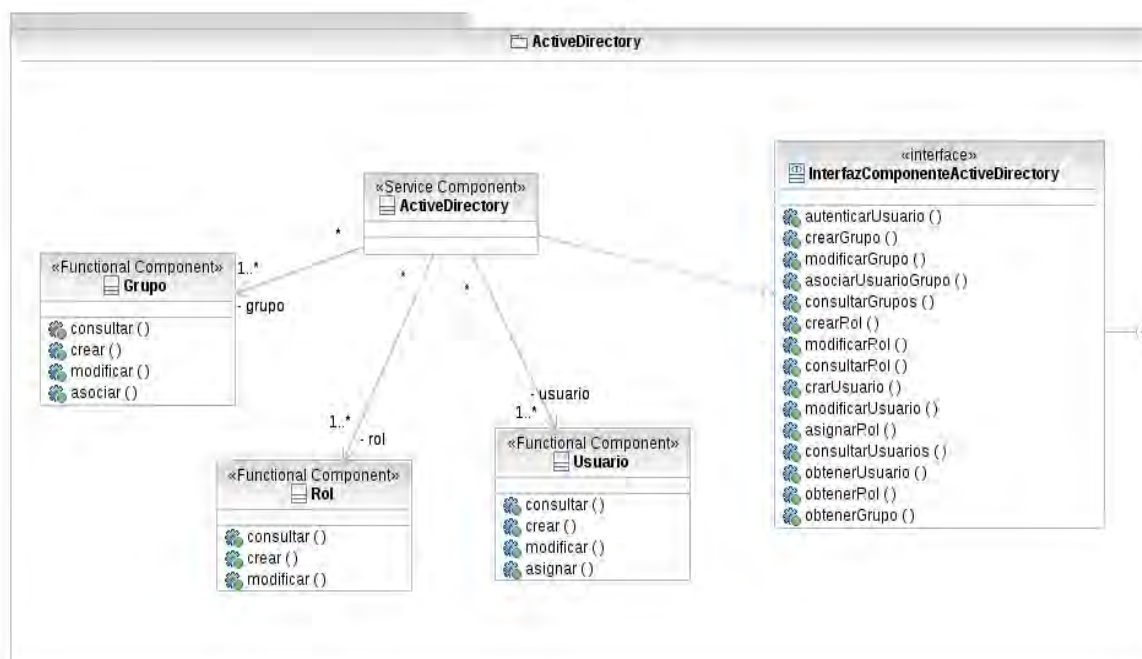
Esta sección describe las responsabilidades funcionales del subsistema “*ActiveDirectory*”.

**Tabla 4: Descripción del Subsistema ActiveDirectory**

<b>Función</b>	<b>Descripción</b>
<i>Realizar la autenticación de usuarios</i>	<i>Proveer un mecanismo para que los usuarios puedan autenticarse con sus credenciales utilizadas en la organización.</i>
<i>Consultar datos de un usuario, Rol o Grupo</i>	<i>Proveer un mecanismo para que sea posible recuperar los datos de usuario, Roles o grupos almacenados en el repositorio centralizado de la organización.</i>

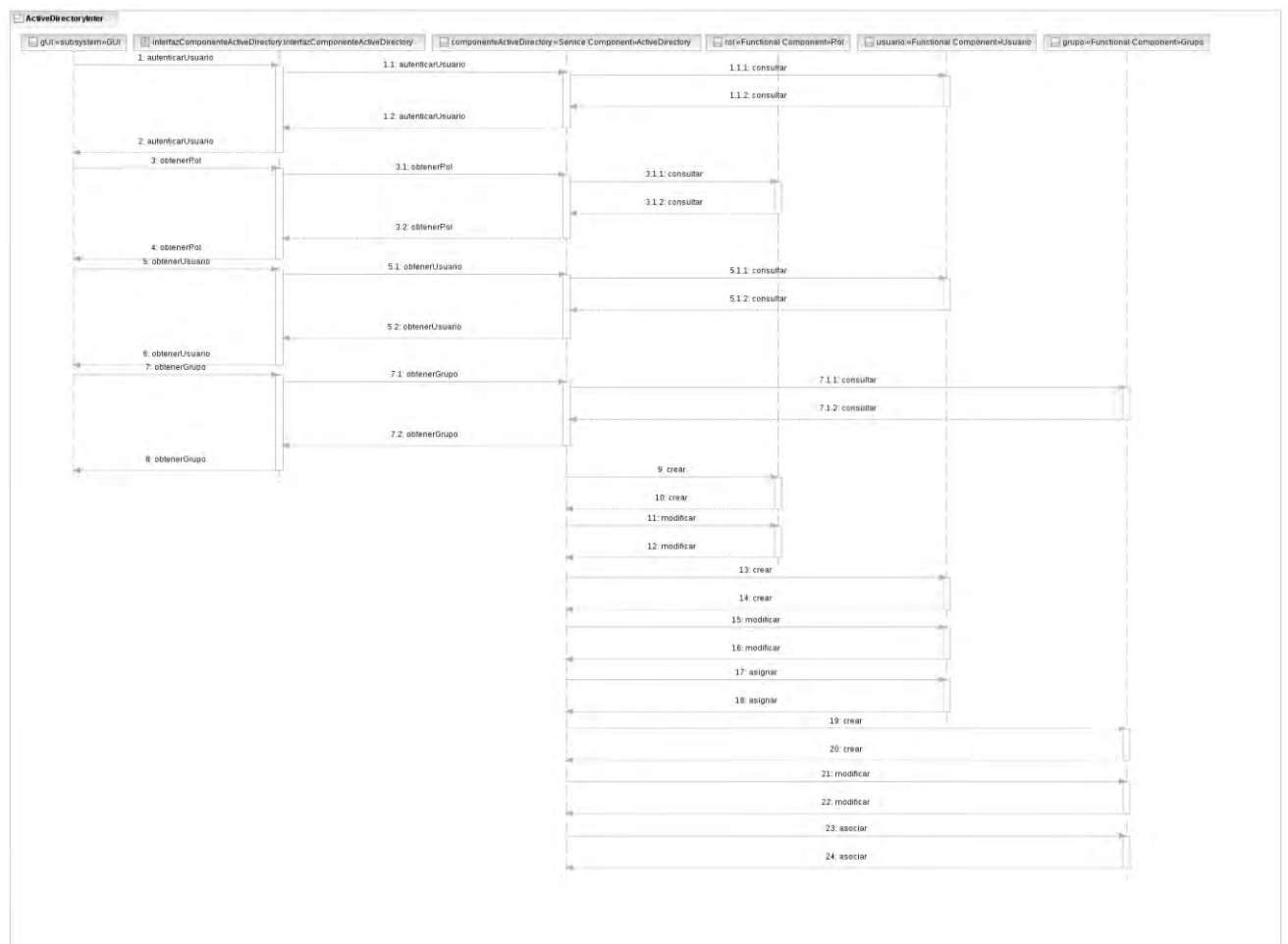
#### 5.2.3.2 Composición e interfaces del Subsistema

La Figura 8 representa el diagrama de clases mostrando los componentes funcionales, técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El “*ComponenteActiveDirectory*”, es responsable de proveer un mecanismo para que los usuarios pertenecientes a la organización, puedan autenticarse utilizando sus credenciales para acceder a las funcionalidades provistas por el “*Business Space*” y de esta forma interactuar con el proceso “*Solicitud de TDC*”.



**Figura 5.8: Composición del subsistema ActiveDirectory**

La Figura 9 representa un diagrama de secuencia que muestra una vista general de la interacción de los componentes que conforman el subsistema “ActiveDirectory”.



**Figura 5.9: Interacción de componentes**

### 5.2.3.3 Comportamiento del Subsistema

El subsistema “ActiveDirectory” es utilizado para realizar la autenticación de los usuarios al momento de ingresar al “Business Space” para interactuar con el proceso de Solicitud de TDC.

Al momento de realizar el ingreso a su espacio de trabajo personal en el “Business Space”, se invoca este componente para verificar si las credenciales ingresadas son correctas y listar en su espacio personal las tareas pendientes del usuario autenticado. Este listado se efectúa automáticamente una vez exitosa la autenticación y utilizando el nombre de usuario provisto por el componente, se listan únicamente las tareas pendientes que tiene el usuario en particular. Adicional a la autenticación de usuarios, este subsistema permitirá el enrutado de las actividades pendientes por atención a los distintos usuarios registrados dependiendo de su rol.

Active Directory de Microsoft es un directorio que almacena componentes pertenecientes a una red, permitiendo identificarlos de forma única y listar sus características. Entre estos componentes destacan para el fin de este subsistema, usuarios, grupos y roles. Estos componentes u objetos se caracterizan por tener un nombre y una serie de atributos descriptivos como por ejemplo descripción, nombre completo de usuario, clave de acceso entre otros. Este componente se basa en el protocolo LDAP (Lightweight Directory Access Protocol) para brindar sus funcionalidades. Este protocolo define la forma como clientes y servidores intercambian la información almacenada en el directorio.

## 5.2.4 Subsistema **BRMS**

### 5.2.4.1 Descripción del Subsistema.

Esta sección describe las responsabilidades funcionales del subsistema BRMS.

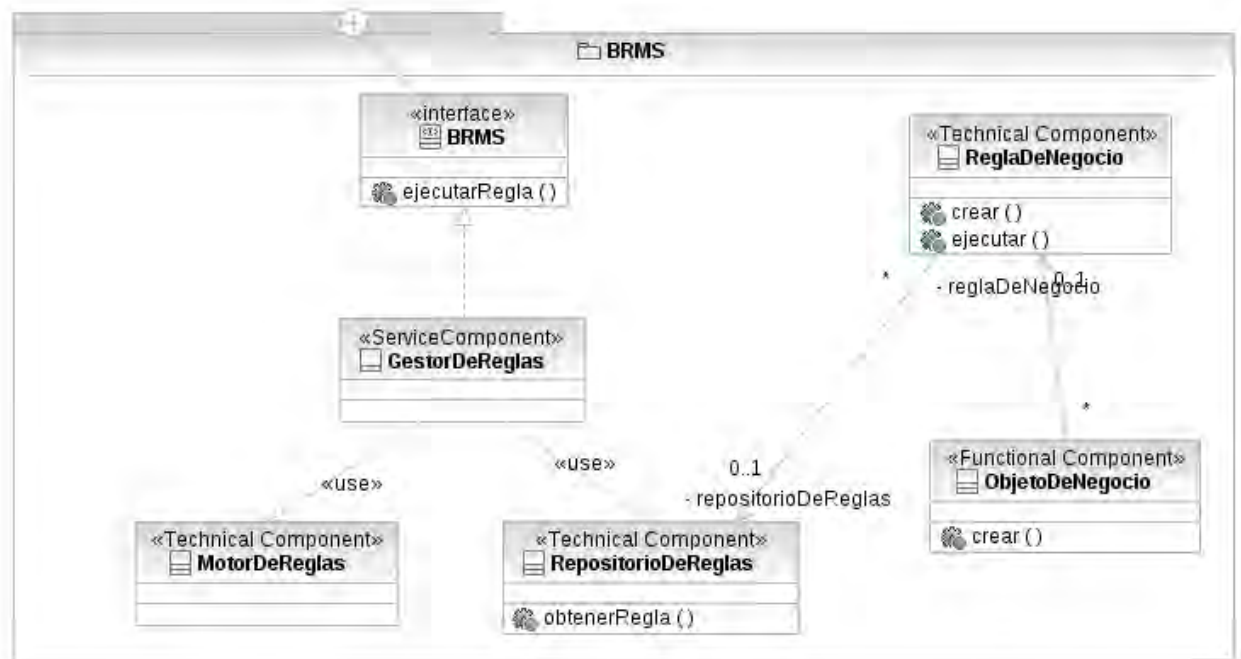
Función	Descripción
<i>Ejecutar Reglas de</i>	<i>Este subsistema mantendrá de forma centralizada todas las reglas</i>

<i>Negocio</i>	<i>de negocio de la solución SOA/BPM, las ejecuta y entrega los resultados al proceso de negocio “Solicitud de TDC” para las decisiones dependientes de los resultados de estas reglas.</i>
----------------	---

**Tabla 5: Descripción del Subsistema BRMS**

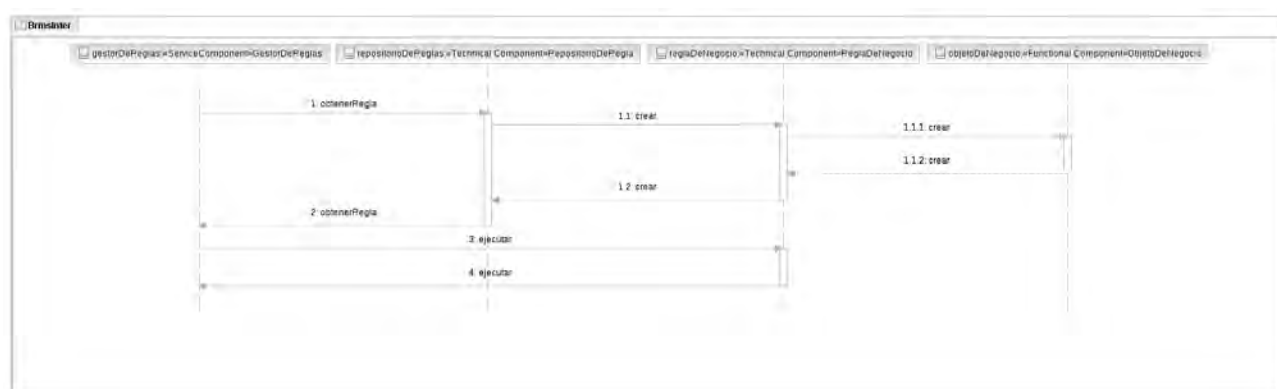
#### 5.2.4.2 Composición e interfaces del Subsistema

La Figura 10 representa el diagrama de clases mostrando los componentes funcionales, técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El subsistema BRMS, es responsable de proveer y ejecutar las reglas de negocio requeridas para la toma de decisiones en el flujo del proceso de negocio “Solicitud de Tarjeta de Crédito”.



**Figura 5.10: Composición del subsistema BRMS**

La Figura 11 representa un diagrama de secuencia que muestra una vista general de la interacción de los componentes que conforman el subsistema “BRMS”.



**Figura 5.11: Interacción de componentes**

### 5.2.4.3 Comportamiento del Subsistema

WebSphere Process Server incluye un editor y motor de reglas de negocio que puede implementar algunas tablas de decisiones y otras reglas simples, y permite poca interacción con el usuario de negocio.

Las reglas de negocio serán provistas como servicios para el proceso de negocio “Solicitud de Tarjeta de Crédito”. Existen diferentes protocolos para poder realizar los llamados a estos servicios, en la solución SOA/BPM para se estarán accediendo a las reglas de negocio mediante WebServices.



## 5.2.5 Subsistema **Solicitud de TDC**

### 5.2.5.1 Descripción del Subsistema.

Esta sección describe las responsabilidades funcionales del subsistema “*Solicitud de TDC*”.

<b>Función</b>	<b>Descripción</b>
<i>Crear Tarea de Emisión Solicitud de TDC</i>	<i>Se crea una nueva tarea con los parámetros necesarios para iniciar una instancia de la tarea Humana que emite una TDC. Al iniciar una instancia de esta tarea humana, se dispara de forma automática el proceso de solicitud de TDC.</i>
<i>Buscar Solicitud por Cédula del cliente</i>	<i>Para poder realizar la atención de un cliente al momento de recibir los recaudos, es necesario contar con un mecanismo que permita recuperar y mostrar los datos de una solicitud a partir de la cédula ingresada.</i>
<i>Registrar la grabación de una TDC</i>	<i>Una vez grabada una TDC, el proceso continúa su ejecución normal. Se provee un mecanismo para que exista la posibilidad de indicarle a la instancia del proceso que la grabación de la TDC fue realizada exitosamente o no.</i>
<i>Registrar distribución de una TDC</i>	<i>Una vez realizada la distribución, el proceso debe continuar su ejecución normal. Se provee un mecanismo para que exista la posibilidad de indicarle a la instancia del proceso que una TDC fue entregada a su destinatario o no.</i>
<i>Gestionar tareas humanas</i>	<i>Se provee un mecanismo que permita a través del subsistema “GUI” gestionar las tareas humanas; tanto las plantillas definidas durante la coreografía, como las instancias activas de estas plantillas.</i>
<i>Gestionar procesos</i>	<i>Se provee un mecanismo que permite a través del subsistema “GUI” gestionar los procesos; tanto las plantillas de los procesos definidos</i>

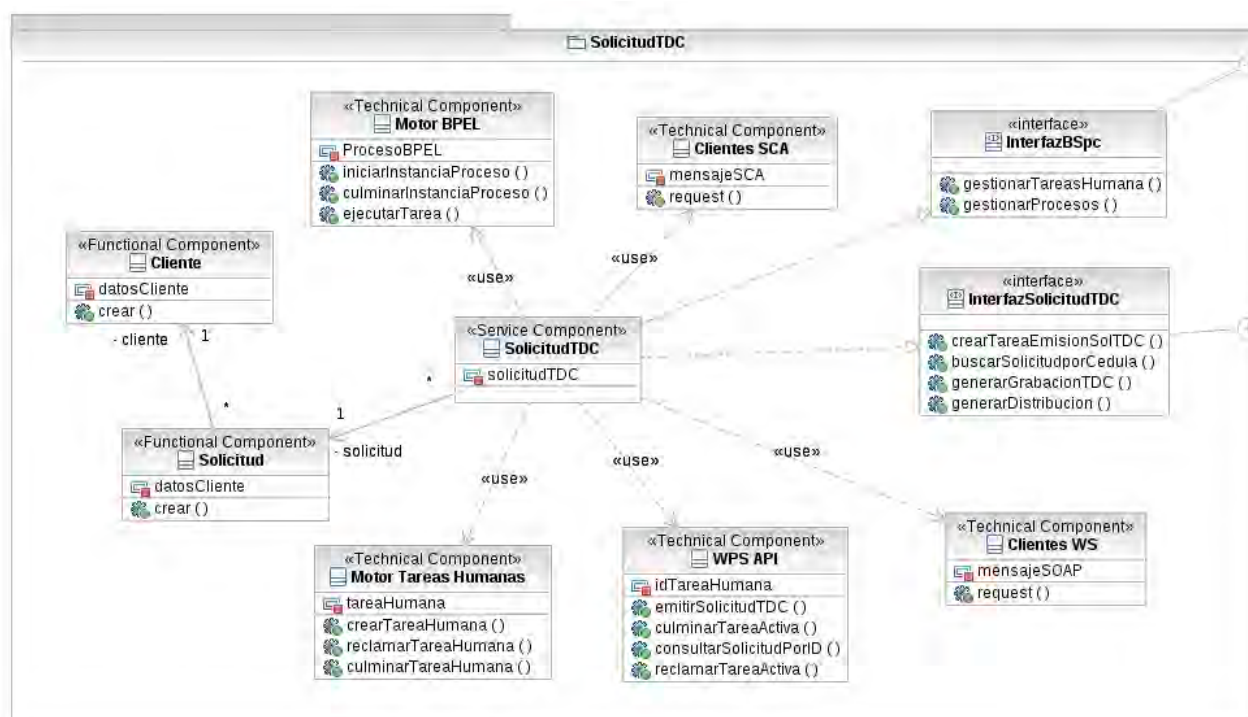
	<i>durante la coreografía, como las instancias activas de estas plantillas.</i>
--	---

**Tabla 6: Descripción del Subsistema Solicitud de TDC**

### 5.2.5.2 Composición e interfaces del Subsistema

La Figura 13 representa el diagrama de clases mostrando los componentes funcionales, técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El subsistema “Solicitud TDC”, provee mecanismos que permiten enviar los datos de una solicitud e iniciar el proceso de Solicitud de TDC. Adicionalmente, provee un mecanismo que permite que una vez realizada la grabación de una TDC, se notifique al proceso de solicitud asociado a esta TDC, el resultado de la grabación. Para la distribución de las tarjetas, también se provee un mecanismo que permite indicarle a los procesos asociados a las tarjetas distribuidas, si estas fueron entregadas a su destinatario final o no. Como último, este subsistema permitirá la consulta de una solicitud en proceso a partir del número de cédula del cliente asociado a esa solicitud.

Para realizar la interacción con la capa superior, este subsistema provee 2 funciones adicionales, expuestas a través de su interfaz “InterfazBSpC” como se puede observar en la figura 13. Los servicios provistos a la capa superior permiten en primera instancia realizar la gestión de las plantillas de las tareas humanas definidas durante la orquestación del proceso de Solicitud de TDC. Igualmente permiten gestionar las instancias activas de estas plantillas. En segundo lugar, permiten gestionar las plantillas de los procesos BPEL definidos durante la orquestación del proceso de Solicitud de TDC. De igual forma que las tareas humanas, este subsistema provee un mecanismo para gestionar las instancias activas de los procesos BPEL generados a partir de estas plantillas.



**Figura 5.12: Composición del subsistema SolicitudTDC**

### 5.2.5.3 Comportamiento del Subsistema

El punto de entrada a este Subsistema es la creación de una nueva Solicitud de TDC. Este subsistema provee una interfaz a través de la cual se podrá realizar esta tarea. Al invocar esta interfaz, se crea una nueva instancia del componente funcional “Solicitud” y se invoca el API de “Websphere Process Server”. A través del API, se crea una nueva instancia de la tarea utilizada para emitir una nueva solicitud. Como se trata de una tarea de invocación, esta recibe la nueva solicitud e inicia una nueva instancia del proceso Solicitud de TDC. En este punto el motor de ejecución BPEL, realiza la orquestación de los servicios involucrados en el proceso de Solicitud TDC, deteniéndose en las tareas humanas para esperar el ingreso de datos por parte de su responsable. Una vez que el proceso llega a una tarea humana, debe esperar a que el responsable de esta tarea la ejecute y esta se marque como completada. En este momento el motor de tareas humanas completa la tarea y permite al motor BPEL continuar con la ejecución del proceso.

Adicional a esto, se requiere consultar todas las solicitudes de TDC pendientes filtrando por el número de cédula del cliente que realizó la solicitud u por otros criterios de búsqueda. Para este fin, el subsistema invoca el API de “Websphere Process Server” para realizar consultas sobre los procesos que se encuentran pendientes y los datos que contienen y devuelve el identificador único de la tarea humana buscada.

Un hito importante durante la solicitud de TDC es la grabación de la tarjeta física. Este es un proceso que ocurre fuera de este subsistema. Dado a que es un hito importante del proceso, se debe registrar si una tarjeta solicitada por un cliente fue grabada o no. Esto se realiza utilizando una tarea humana que obliga al proceso a esperar una entrada externa para continuar. El subsistema invocará el API de “Websphere Process Server” para ubicar la tarea humana que se encuentra a la espera del resultado de la grabación. Una vez ubicada la tarea humana correspondiente, se reclama (“claim”) al motor de tareas humanas y se completa enviando como parámetro si la tarjeta fue grabada o no. Al completarse la tarea humana, el proceso continúa su ejecución normal.

El último hito del proceso de solicitud de TDC es la distribución de la tarjeta solicitada por un cliente. Se debe proveer un mecanismo para incluir en el proceso de una solicitud el resultado de la distribución, de la misma forma que la grabación de la TDC. Nuevamente el subsistema realiza una búsqueda por un identificador a través del API de “Websphere Process Server” para ubicar la tarea humana que se encuentra a la espera del resultado de la distribución. Una vez ubicada, la tarea humana, se reclama (“claim”) al motor de tareas humanas para que ya no esté disponible para otros usuarios y se completa enviando los datos obligatorios de forma automática. Una vez completada la tarea, el proceso invocado por la solicitud de una TDC culmina de forma satisfactoria.

## Subsistema **Enterprise Service Bus**

### 5.2.5.4 Descripción del Subsistema.

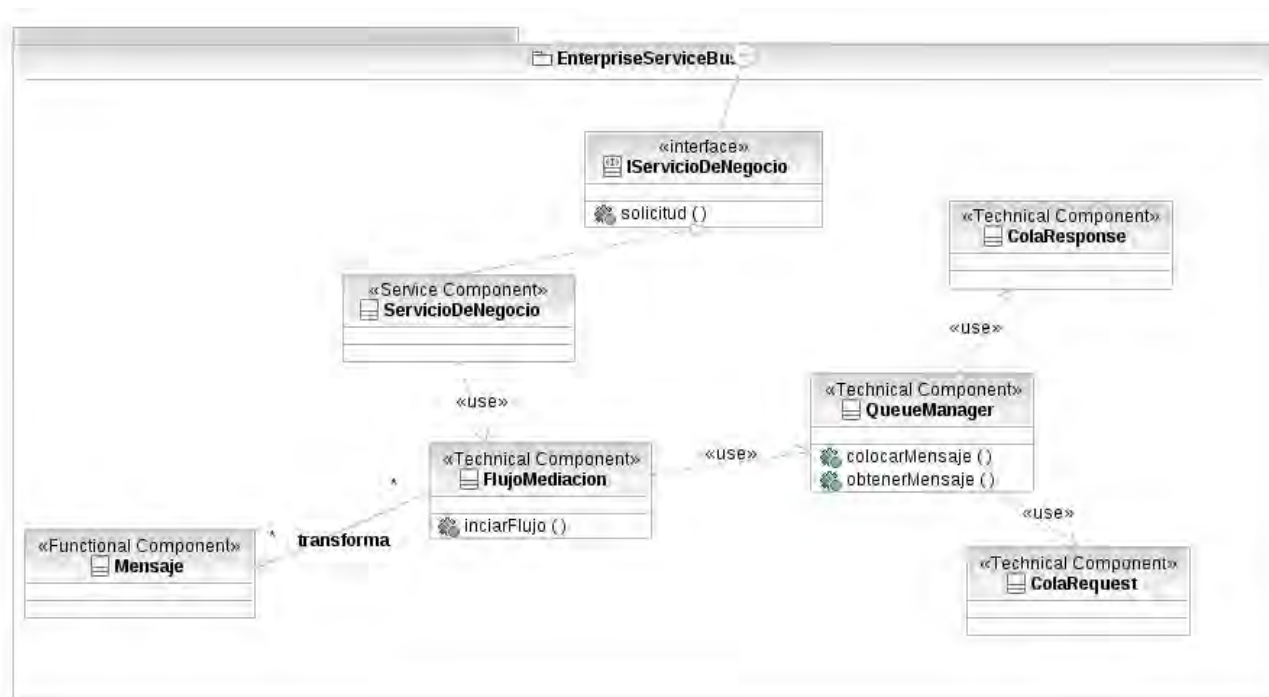
Esta sección describe las responsabilidades funcionales del subsistema Enterprise Service Bus.

<b>Función</b>	<b>Descripción</b>
<i>Proveer Servicios de negocio</i>	<i>Proveer todos los servicios de negocio requeridos para desarrollar el flujo del proceso de negocio “Solicitud de Tarjeta de Crédito”.</i>
<i>Orquestación de servicios</i>	<i>Orquestar Servicios atómicos para proveer servicios compuestos de acuerdo a los requerimientos del negocio.</i>
<i>Mediación y transformación de mensajes</i>	<i>Transformar los mensajes a los formatos requeridos para cada uno de los destinos de los flujos de mediación.</i>

**Tabla 7: Descripción del Subsistema Enterprise Service Bus**

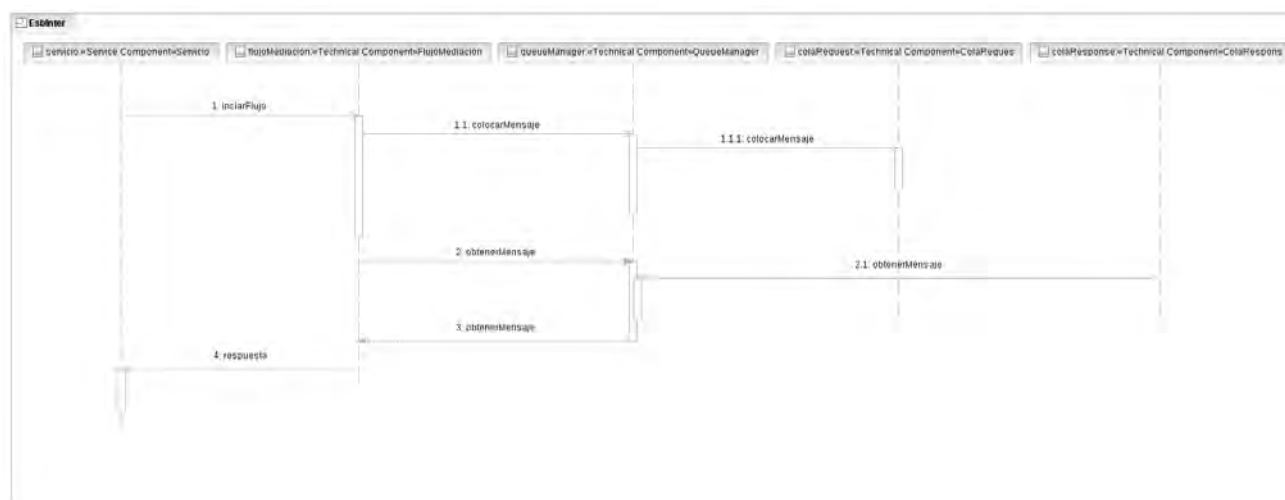
### 5.2.5.5 Composición e interfaces del Subsistema

La Figura 17 representa el diagrama de clases mostrando los componentes funcionales, técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El subsistema Enterprise Service Bus, es responsable de proveer todos los servicios de negocio requeridos para desarrollar el flujo del proceso de negocio “Solicitud de Tarjeta de Crédito” y mediar con los sistemas legados con el uso de colas MQ para enviar y recibir los mensajes a las solicitudes hechas contra estos legados. Igualmente el componente Enterprise Service Bus realizará las transformaciones requeridas a los mensajes hacia los formatos requeridos por las plataformas destino.



**Figura 5.13: Composición del subsistema Enterprise Service Bus**

La Figura 18 representa un diagrama de secuencia que muestra una vista general de la interacción de los componentes que conforman el subsistema “Enterprise Service Bus”.



**Figura 5.14: Interacción de componentes**

### 5.2.5.6 Comportamiento del Subsistema

El subsistema “Enterprise Service Bus” es implementado mediante el producto WebSphere Message Broker. El Message Broker es un poderoso intermediario de información con el cual la data del negocio pueda fluir entre aplicaciones y plataformas totalmente dispares en forma de mensajes. Diferentes reglas pueden ser aplicadas a los mensajes durante el flujo para poder enrutar, transformar o recibir información como sea requerido.

El Message Broker en la arquitectura definida tendrá (tal como indicado por el nombre del componente) el rol de un bus de servicios empresarial, actuando como mediador entre los consumidores y los proveedores de servicios (En el contexto del proyecto SOA/BPM, WebSphere Process Server es el consumidor de los servicios). El message broker desacopla el consumidor de servicio del proveedor, entregando servicios para resolver diferencias en protocolo y formato. Con esto obtendremos múltiples beneficios tales como:

- Reducir el número, tamaño y complejidad de las interfaces.
- Reducir el impacto de generar cambios de ubicación de servicios o formatos.
- Permite la integración de recursos dispares
- Permite cambiar un proveedor de servicio por uno nuevo, sin la necesidad de tener que afectar al consumidor.

En la arquitectura de la solución SOA/BPM el Websphere Message Broker utilizará un paradigma de mensajería de Request/Response para realizar solicitudes a los sistemas legados transformando el mensaje de la solicitud en el formato requerido por el programa RPG destino y colocándolo en una cola de “Request”. La respuesta será colocada por el legado en una cola de “Response”, lo cual desencadenará un flujo de mediación para transformar el mensaje al formato esperado por el WebSphere Process Server, consumidor de los servicios expuestos en forma de Web Services en la arquitectura planteada para la solución SOA/BPM.



## 5.2.6 Subsistema DispatcherPool

### 5.2.6.1 Descripción del Subsistema.

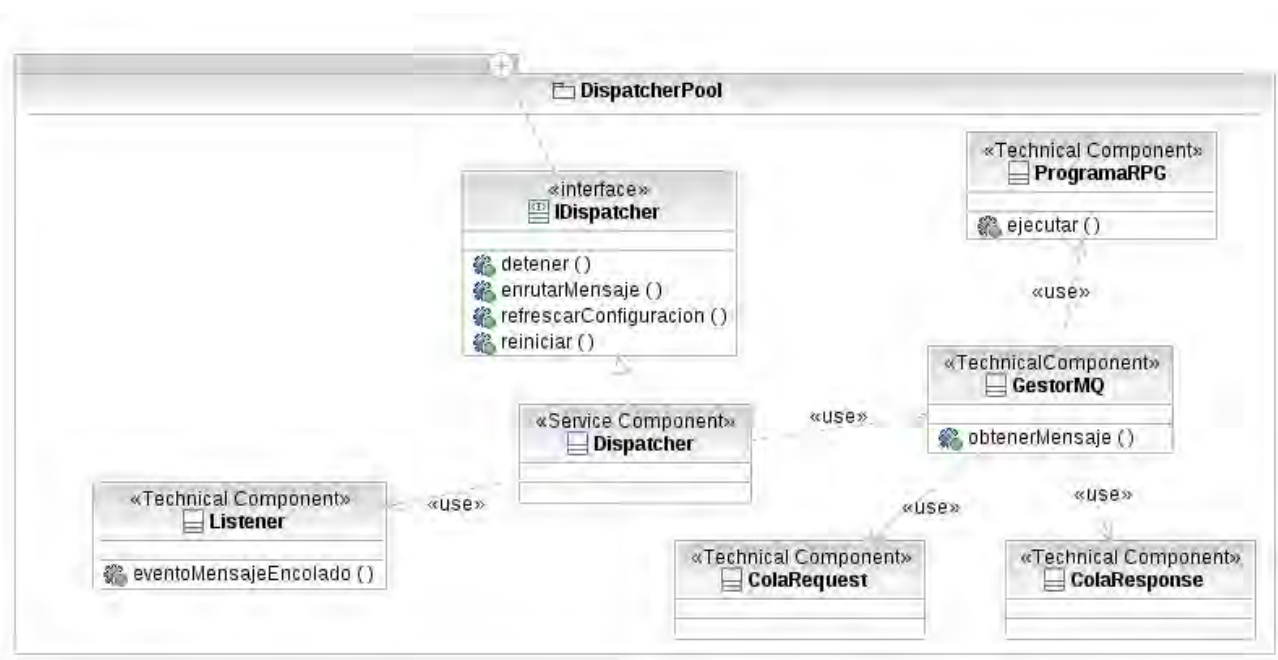
Esta sección describe las responsabilidades funcionales del subsistema DispatcherPool.

Función	Descripción
<i>Enrutar las solicitudes hechas a los programas RPG responsables</i>	<i>El dispatcher obtiene las solicitudes realizadas a los legados a través de un mensaje MQ en la cola de “Request”, delega el mensaje al programa RPG responsable de ejecutar la solicitud, y al obtener la respuesta coloca la misma en la cola de “Response”.</i>

**Tabla 8: Descripción del Subsistema DispatcherPool**

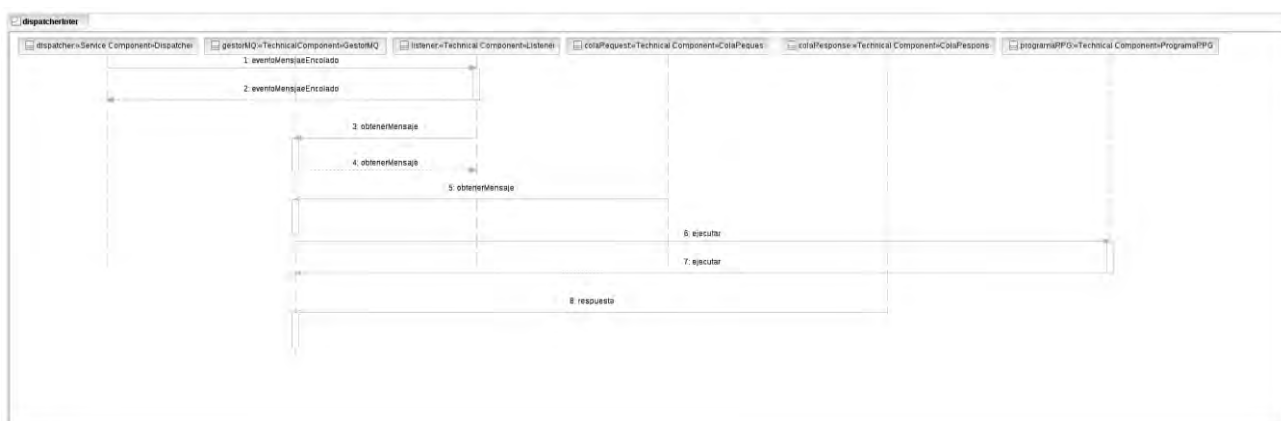
### 5.2.6.2 Composición e interfaces del Subsistema

La Figura 19 representa el diagrama de clases mostrando los componentes funcionales, técnicos y de servicio relevantes para describir la composición del subsistema y sus interfaces para exponer su funcionalidad. El subsistema DispatcherPool, esta compuesto de un pool de componentes “Dispatcher” que tendrán como responsabilidad el enrutamiento de las solicitudes realizadas hacia los legados, a los programas RPG indicados para resolver la solicitud. Estas solicitudes llegan al dispatcher en forma de mensajes MQ y el dispatcher obtiene del mensaje MQ los parámetros con el formato requerido para el programa destino e invoca su ejecución. Finalmente el dispatcher obtiene la respuesta y coloca el mensaje en la cola de “Response” para retornar la respuesta.



**Figura 5.15:** Composición del subsistema DispatcherPool

La Figura 20 representa un diagrama de secuencia que muestra una vista general de la interacción de los componentes que conforman el subsistema “DispatcherPool”.



**Figura 5.16:** Interacción de componentes

### 5.2.6.3 Comportamiento del Subsistema

El DispatcherPool se compone de un pool de componentes Dispatcher, desarrollados en RPG. Estos “Dispatcher” utilizarán un paradigma de mensajería “Request”, “Response” para responder a todas las solicitudes realizadas hacia los legados. Los mensajes con la solicitud

serán transmitidos a través de una cola MQ (cola “Request”). El mensaje MQ contiene en el encabezado del mensaje, la metadata para enrutar la solicitud hacia el programa RPG adecuado. El dispatcher debe remover el encabezado MQ y delegar el procesamiento de la solicitud al programa RPG, conservando la información del encabezado (El ReplyIdentifier) para poder generar la respuesta con el resultado de la ejecución del programa RPG. Finalmente el dispatcher, debe generar un mensaje MQ de respuesta, colocar en el mensaje la metadata para poder responder a la solicitud, la respuesta del programa RPG en el body del mensaje y colocar este mensaje MQ en la cola de respuesta (Cola “Response”).

La metadata será colocada en el body del mensaje como la primera parte de la trama de solicitud. Cada mensaje enviado y tendrá el siguiente formato:

**ID de sesión: Char(30)**

**ID usuario: Char(15)**

**ID programa: Char(10)**

**Versión: Char(4)**

**Cantidad de tramas en response: Char(4)**

---

## 5.3 Atributos de Componentes

### 5.3.1 Componentes para Subsistema ComponenteSolicitudTDC

<b>Tipo de Componente:</b> Servicio <b>Nombre:</b> ComponenteSolicitudTDC	
<b>Descripción</b>	Este componente expone las funcionalidades del subsistema. Su responsabilidad es proveer los servicios para el envío de la solicitud y generar una versión imprimible de la solicitud para el cliente solicitante; para esta funcionalidad este componente se vale de los servicios del subsistema ReporteSolicitud.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>solicitudTDC</li> </ul>

<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Este componente provee la interfaz para solicitud de TDC a usuarios clientes y no clientes del Banco.</li> <li>• Data requerida para representar el formulario debe acceder a la data en un repositorio local y no acceder a los legados en la intranet</li> <li>• La implementación del componente se hará mediante el uso de J2EE</li> <li>• Despliegue del componente se realizara en el servidor de Internet banking.</li> </ul>
<b>Reglas</b>	<ul style="list-style-type: none"> <li>• Data representada en el formulario depende del tipo de la solicitud de TDC</li> </ul>
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• enviarSolicitud</li> <li>• generarReporteSolicitude</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• generarReporteSolicitud (del subsistema ReporteSolictud)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> Cliente	
<b>Descripción</b>	Componente funcional para almacenar los datos del solicitante.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• id</li> <li>• nombres</li> <li>• apellidos</li> <li>• sexo</li> <li>• profesion</li> <li>• edad</li> <li>• estadoCivil</li> <li>• correo</li> <li>• cargasFamiliares</li> <li>• telefonos</li> <li>• idConyugue</li> <li>• datosVivienda</li> </ul>

	<ul style="list-style-type: none"> <li>• automovil</li> <li>• datosTrabajo</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Este componente contiene todos los datos relacionados directamente al cliente</li> </ul>
<b>Reglas</b>	<ul style="list-style-type: none"> <li>• Toda solicitud contiene un componente de este tipo para representar al cliente</li> </ul>
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> SolicitudTDC	
<b>Descripción</b>	Componente funcional para almacenar los datos de la solicitud de TDC.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• tipoSolicitud</li> <li>• tipoProducto</li> <li>• cliente</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Este componente contiene todos los datos requeridos para todos los tipos de solicitud contemplados en la solución (Clásica, Empresarial, Visa joven y Suplementaria)</li> </ul>
<b>Reglas</b>	<ul style="list-style-type: none"> <li>• Los datos requeridos para la solicitud varían dependiendo del tipo de solicitud.</li> </ul>
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
-----------------------------	---

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> WebService Clients	
<b>Descripción</b>	Componente técnico responsable para hacer los llamados al WS en Process Server para realizar el envío de la solicitud e iniciar el proceso de negocio.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>solicitudTDC</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>Los llamados al WebService para crear la instancia de un nuevo proceso deben manejar WS-Security para presentar credenciales para el llamado al servicio</li> <li>Debe existir un usuario de servicio, para poder realizar una autenticación exitosa, ya que este servicio esta habilitado para usuarios no clientes del BE.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

### 5.3.2 Componentes para Subsistema GUI

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> Widget	
<b>Descripción</b>	Componente técnico que representa las interfaces graficas que provee el BS al a solución.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>Todos los componentes gráficos para interactuar con el proceso de negocio para la solicitud de TDC son componentes de este tipo.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> Human Task Widget	
<b>Descripción</b>	Componente técnico que representa una interfaz gráfica para interactuar con las tareas humanas del proceso de solicitud de TDC. Este componente hereda todas las características de “Widget”.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>Todos los Widgets podrán ser incluidos al espacio de trabajo en Business Space por usuarios autorizados</li> </ul>
<b>Reglas</b>	

<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> Task Widget	
<b>Descripción</b>	Componente técnico que representa la interfaz grafica con el listado/búsqueda de tareas disponibles, que permite solicitar alguna de estas tareas para ser ejecutada por el usuario autenticado. Este componente hereda todas las características de “Widget”.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Este Widget debe estar disponible por defecto para todos los perfiles autorizados en BS.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>



### 5.3.3 Componentes para Subsistema ActiveDirectory

<b>Tipo de Componente:</b> Servicio <b>Nombre:</b> ActiveDirectory	
<b>Descripción</b>	Este componente permite autenticar la validez de un usuario a través del envío de credenciales (usuario y password), y adicionalmente validar los permisos de un usuario autenticado.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Este Widget debe estar disponible por defecto para todos los perfiles autorizados en BS.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• autenticarUsuario</li> <li>• crearGrupo</li> <li>• modificarGrupo</li> <li>• asociarUsuariosGrupo</li> <li>• consultarGrupos</li> <li>• crearRol</li> <li>• modificarRol</li> <li>• crearUsuario</li> <li>• modificarUsuario</li> <li>• asignarlo</li> <li>• consultarUsuarios</li> <li>• obtenerUsuario</li> <li>• obtenerRol</li> <li>• obtenerGrupos</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> Grupo	
<b>Descripción</b>	Componente funcional que representa una agrupación lógica de usuarios para proveer una determinada funcionalidad.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> Rol	
<b>Descripción</b>	Componente funcional que representa la función del usuario en la estructura jerárquica del directorio de usuarios.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>

<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> Usuario	
<b>Descripción</b>	Componente funcional que representa en el sistema a los usuarios de la aplicación
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Id</li> <li>• Nombre</li> <li>• Apellido</li> <li>• Rol</li> <li>• Grupos</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

### 5.3.4 Componentes para Subsistema BRMS

<b>Tipo de Componente:</b> Servicio <b>Nombre:</b> GestorDeReglas	
<b>Descripción</b>	Componente de servicio que provee los servicios para la invocación de las reglas de negocio.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Todas las reglas de negocio contempladas para la solución deben ser implementadas utilizando este componente.</li> <li>• Las reglas de negocio van a ser expuestas como WebServices</li> <li>• Todos los WebServices deben manejar WS-Security para los llamados a todos los servicios de reglas de negocio.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> MotorDeReglas	
<b>Descripción</b>	Componente técnico que provee el ambiente de ejecución para las reglas de negocio desarrolladas.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Todas las reglas de negocio contempladas para la solución deben ser ejecutadas en este motor de reglas</li> </ul>

<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> RepositorioDeReglas	
<b>Descripción</b>	Componente técnico que funciona como repositorio para las reglas de negocio desarrolladas por el ambiente de BRMS.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Todas las reglas de negocio contempladas para la solución deben estar almacenadas y versionadas en este repositorio en este repositorio.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> ObjetoDeNegocio	
<b>Descripción</b>	Componente técnico que representa un objeto de negocio requerido para la generación de reglas de negocio desarrolladas en el sistema de BRMS.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• (Definidos por cada OBJ de negocio)</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> ReglaDeNegocio	
<b>Descripción</b>	Componente técnico que representa un artefacto que implementa una regla de negocio desarrollada en el sistema de BRMS.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Regla</li> <li>• Tablas de decisión</li> <li>• Árboles de decisión</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Todas las reglas de negocio contempladas para la solución deben estar desarrolladas como un artefacto de reglas de negocio.</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>

<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

### 5.3.5 Componentes para Subsistema Solicitud de TDC

<b>Tipo de Componente:</b> Servicio <b>Nombre:</b> SolicitudTDC	
<b>Descripción</b>	Este componente de servicio representa el proceso de negocio para la solicitud de TDC.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Flujos</li> <li>• Subflujos</li> <li>• Tareas</li> <li>• Tareas Humanas</li> <li>• Servicios</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Las instancias del proceso se crean a través de una aplicación web externa, cuando los usuarios generan una solicitud.</li> <li>• Se debe proveer un servicio, para que aplicaciones externas al Process Server puedan instancias el proceso de negocio.</li> <li>• El servicio para generar la instancia del proceso será expuesto como un WebService que debe manejar su seguridad vía WS-Security</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• gestionarTareaHumana</li> <li>• gestionarProceso</li> <li>• crearTareaSolicitudDeTDC</li> <li>• buscarSolicitudPorCedula</li> <li>• generarGrabacionTDC</li> <li>• generarDistribucion</li> </ul>

<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
-----------------------------	---

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> ClientesSCA	
<b>Descripción</b>	Componente técnico para realizar los llamados a servicios SCA. En el contexto de la solución, llamados desde WPS a servicios desplegados en el WMB o llamadas a servicios locales al WPS.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>mensajeSCA</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> Clientes WS	
<b>Descripción</b>	Componente técnico que tiene la responsabilidad de llamados a WS externos al Process Server.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>Mensaje SOAP</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>



<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> Motor BPEL	
<b>Descripción</b>	Componente técnico que realizará la interpretación del código BPEL que se desplegará en el Process Server para el flujo del proceso de negocio de la Solicitud de TDC.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Proceso Solicitud de TDC</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> Motor Tareas Humanas	

<b>Descripción</b>	Componente técnico dentro del subsistema SolicitudTDC implementado en WPS que se encargara de gestionar las tareas humanas contempladas para la interacción con el proceso de negocio.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Proceso Solicitud de TDC</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> WPS API	
<b>Descripción</b>	Componente técnico dentro del subsistema SolicitudTDC implementado en WPS que permitirá la interacción programática con los procesos de negocio instanciados.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• idTareaHumana</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

### 5.3.6 Componentes para Subsistema Enterprise Service Bus

<b>Tipo de Componente:</b> Servicio <b>Nombre:</b> ServicioDeNegocio	
<b>Descripción</b>	Componente de servicio que expone al repositorio de servicios los servicios empresariales contemplados para el proceso de negocio de Solicitud de Tarjeta de Crédito.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Interfaz de servicio</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Todos los servicios de negocio serán expuestos como WebServices desde este componente</li> <li>• Todos los Web Services expuestos deben proveer un mecanismo de autenticación a través de WS-Security</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• Solicitud</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> FlujoMediacion	
<b>Descripción</b>	Componente técnico encargado de las mediaciones y transformaciones de los mensajes que intercambiarán el proceso de negocio con los legados del Banco.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Mensaje</li> <li>• nodo</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>

<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Funcional <b>Nombre:</b> Mensaje	
<b>Descripción</b>	Componente Funcional, que portara toda la data desde los legados al proceso de negocio y desde el proceso de negocio hacia los legados.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• Encabezado</li> <li>• Body</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• El body del mensaje contiene metadata para el enrutamiento del mensaje, cuando el destinatario es el legado (dispatcher)</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> QueueManager
--

<b>Descripción</b>	Componente Técnico encargado de almacenar y gestionar las colas requeridas para el pase de mensajes entre procesos y legados del banco.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>cola</li> </ul>
<b>Políticas</b>	
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> ColaResponse	
<b>Descripción</b>	Componente Técnico que se encargará de transmitir las respuestas de los legados hacia el proceso de negocio
<b>Atributos</b>	<ul style="list-style-type: none"> <li></li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>Cada servicio tiene que tener asignada una cola de response para que el legado pueda responder al componente Enterprise Service Bus con el mensaje de respuesta</li> </ul>
<b>Reglas</b>	
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

<b>provee</b>	
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> ColaRequest	
<b>Descripción</b>	Componente Técnico que se encargará de transmitir las solicitudes del proceso de negocio hacia los legados.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>•</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>• Cada servicio tiene que tener asignada una cola de request para poder realizar solicitudes a los legados del Banco.</li> </ul>
<b>Reglas</b>	N/A
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>• N/A (Solo para componentes de servicio)</li> </ul>

### 5.3.7 Componentes para Subsistema <DispatcherPool>

<b>Tipo de Componente:</b> Servicio <b>Nombre:</b> Dispatcher	
<b>Descripción</b>	Componente de servicio que se encargara de enrutar los mensajes a los programas RPG adecuados para responder a una solicitud realizada por una ColaRequest.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>• mensaje</li> </ul>

<b>Políticas</b>	<ul style="list-style-type: none"> <li>Habr� un pool de componentes dispatcher para un conjunto determinado de servicios</li> </ul>
<b>Reglas</b>	N/A
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>Detener</li> <li>RefrescarConfiguracion</li> <li>Reiniciar</li> <li>enrutarMensaje</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

<b>Tipo de Componente:</b> T�cnico <b>Nombre:</b> Listener	
<b>Descripci�n</b>	Componente t�cnico encargado de escuchar los mensajes que llegan a la ColaRequest para enrutar el mensaje al programa RPG destinado a resolver la solicitud enviada por el proceso de negocio.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Reglas</b>	<ul style="list-style-type: none"> <li>Cuando opera un pool de dispatchers, al llegar un mensaje a la ColaResponse, el primer listener disponible debe tomar el mensaje</li> </ul>
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>



<b>Consumidos</b>	
-------------------	--

<b>Tipo de Componente:</b> Técnico <b>Nombre:</b> GestorMQ	
<b>Descripción</b>	Componente técnico encargado de extraer los mensajes de la ColaRequest y escribir las respuestas en la ColaResponse.
<b>Atributos</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Políticas</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Reglas</b>	<ul style="list-style-type: none"> <li>La solicitud siempre llega por un componente ColaResponse</li> <li>La respuesta siempre es escrita en una ColaRequest</li> </ul>
<b>Variaciones</b>	<ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Servicios que provee</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>
<b>Servicios Consumidos</b>	<ul style="list-style-type: none"> <li>N/A (Solo para componentes de servicio)</li> </ul>

---

## 5.4 Abreviaturas

Abreviatura	Significado
WAS	Websphere Application Server
WMB	Webpsphere Message Broker
WPS	Websphere Process Server
WMQ	Websphere Message Queue
TDC	Tarjeta de Credito
MQ	Message Queue
J2EE	Plataforma empresarial de Java, correspondiente a sus siglas en ingles “Java Platform, Enterprise Edition”
EAR	Un formato de empaquetamiento empresarial de archivo basado en la plataforma J2EE (Enterprise Archive)
SOA	Arquitectura Orientada a Servicios
RPG	Lenguaje de programación
RPG/ILE	Última versión del lenguaje de programación RPG
TCP/IP	Protocolo de comunicación usado en internet mediante un identificación denominada IP
BPM	Business Process Management
TDC	Tarjeta de Crédito
BPEL	Business Process Execution Language