

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR



**FACULTAD DE INGENIERÍA
ESCUELA DE SISTEMAS**

**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO DE SISTEMAS**

**“DESARROLLO DE UNA GUÍA TÉCNICA PARA INTERFACES
ENTRE APLICACIONES DE DIFERENTES PLATAFORMAS,
BASADA EN LA ARQUITECTURA ORIENTADA A SERVICIOS
SOA”**

Autor: Juan Pablo Almeida Durán

Directora: Beatriz Campos

Quito, Noviembre de 2012

Agradecimiento

Expreso mi profunda gratitud y admiración a mis padres Eduardo Almeida y Dalinda Durán por sus valiosas enseñanzas y dedicación, también a mi hermana María Isabel Almeida por su respaldo incondicional y su ejemplo durante todos estos años.

De igual manera agradezco a la Ingeniera Beatriz Campos por su acertada dirección en el desarrollo de esta tesis.

Finalmente quiero agradecer a toda mi familia y amigos por todos estos años de soporte y motivación que han ayudado a culminar con éxito este proyecto.

ÍNDICE

| | |
|--|------------|
| <i>I. Resumen</i> | <i>XII</i> |
| <i>II. Introducción</i> | <i>XIV</i> |
| CAPÍTULO I | 1 |
| 1. Arquitectura orientada a servicios SOA. | 1 |
| 1.1. Integración de sistemas informáticos. | 1 |
| 1.1.1. Sistemas de Workflow (flujo de trabajo). | 2 |
| 1.1.2. Sistemas de administración de procesos de negocio..... | 5 |
| 1.1.3. Estrategias de integración de sistemas. | 7 |
| 1.1.4. Arquitectura basada en servicios web. | 9 |
| 1.2. Introducción a la arquitectura orientada a servicios SOA..... | 10 |
| 1.2.1. Formas de implementar SOA en la empresa. | 12 |
| 1.2.2. Estructura de SOA. | 13 |
| 1.3. Principios para el desarrollo, implementación y mantenimiento de SOA. | 18 |
| 1.3.1. Bus de servicio empresarial (ESB). | 18 |
| 1.3.2. Estructura del bus de servicios empresariales (ESB)..... | 20 |
| 1.4. Ciclo de vida y modelo de madurez de la arquitectura orientada a servicios (SOA)..... | 21 |
| 1.4.1. Ciclo de vida de SOA | 21 |
| 1.4.2. Modelo de madurez de SOA. | 22 |
| 1.5. Estándares manejados para la implementación de SOA. | 23 |
| CAPÍTULO II | 26 |
| 2. Herramientas de integración de sistemas, compatibles con SOA. | 26 |
| 2.1. Descripción de la funcionalidad de una interfaz entre aplicaciones..... | 26 |
| 2.1.1. Integración de la capa de presentación..... | 27 |
| 2.1.2. Integración de la capa de datos. | 28 |
| 2.1.3. Integración funcional. | 29 |
| 2.1.3.1. Aproximaciones de la integración funcional..... | 31 |
| 2.2. Funcionamiento de las herramientas para integración de sistemas con la arquitectura SOA. | 34 |
| 2.3. Como realizar integración de sistemas con las herramientas de integración disponibles..... | 37 |
| 2.3.1. Mensajería. | 37 |
| 2.3.2. Conectores. | 38 |
| 2.4. Herramientas disponibles para la realización de interfaces entre aplicaciones utilizando la arquitectura orientada a servicios. | 39 |
| 2.4.1. Herramientas para la realización de interfaces entre aplicaciones utilizando la arquitectura orientada a servicios, bajo el modelo de licenciamiento propietario..... | 39 |
| 2.4.1.1. Microsoft® Biztalk® Server | 40 |
| 2.4.1.2. Oracle® Fusion Middleware® | 40 |
| 2.4.1.3. IBM® WebSphere Software® | 41 |
| 2.4.1.4. SAP NetWeaver® | 42 |
| 2.4.1.5. Informatica® PowerCenter® | 43 |

| | | |
|--|--|-----------|
| 2.4.1.6. | WebMethods® | 44 |
| 2.4.1.7. | TIBCO® Business Integration® | 44 |
| 2.4.1.8. | Sonic ESB® | 45 |
| 2.4.2. | Herramientas para la realización de interfaces entre aplicaciones utilizando la arquitectura orientada a servicios, bajo el modelo de licenciamiento libre. | 46 |
| 2.4.2.1. | Mule ESB Community Edition | 46 |
| 2.4.2.2. | Sopera | 47 |
| 2.4.2.3. | Petals..... | 48 |
| 2.4.2.4. | OpenEAI | 49 |
| 2.4.2.5. | Apache ServiceMix..... | 49 |
| 2.4.2.6. | Apache Camel | 50 |
| 2.5. | Cuadro comparativo de herramientas para la realización de interfaces entre aplicaciones. | 51 |
| CAPÍTULO III..... | | 56 |
| 3. Guía técnica de las herramientas escogidas para interfaces entre aplicaciones y que usan la arquitectura SOA. | | 56 |
| 3.1. | Introducción histórica de las herramientas escogidas para ser analizadas. | 56 |
| 3.1.1. | Microsoft® Biztalk® Server | 56 |
| 3.1.2. | Oracle Fusion Middleware® SOA Suite. | 57 |
| 3.1.3. | Mule ESB Community Edition. | 58 |
| 3.2. | Análisis de requerimientos que poseen las herramientas escogidas para ser usadas de interfaces entre aplicaciones. | 58 |
| 3.2.1. | Microsoft® Biztalk® Server. | 58 |
| 3.2.1.1. | Requerimiento de hardware de Microsoft® Biztalk® Server..... | 59 |
| 3.2.1.2. | Requerimiento de software de Microsoft® Biztalk® Server. | 59 |
| 3.2.2. | Oracle Fusion Middleware® SOA Suite. | 60 |
| 3.2.2.1. | Requerimiento de hardware de Oracle Fusion Middleware® SOA Suite. | 60 |
| 3.2.2.2. | Requerimiento de software de Oracle Fusion Middleware® SOA Suite. | 61 |
| 3.2.3. | Mule ESB® Community Edition. | 62 |
| 3.2.3.1. | Requerimiento de hardware de Mule ESB® Community Edition. | 62 |
| 3.2.3.2. | Requerimiento de software de Mule ESB® Community Edition. | 62 |
| 3.3. | Análisis económico de las herramientas de interfaces entre aplicaciones utilizadas. | 63 |
| 3.3.1. | Análisis económico de Microsoft® Biztalk® Server. | 63 |
| 3.3.2. | Análisis económico de Oracle Fusion Middleware® SOA Suite. | 63 |
| 3.3.3. | Análisis económico de Mule ESB® Community Edition..... | 64 |
| 3.3.4. | Cuadro del análisis de costos de las herramientas. | 64 |
| 3.4. | Análisis detallado del proceso de instalación de las herramientas de interfaces entre aplicaciones. | 64 |
| 3.4.1. | Instalación de Microsoft® Biztalk® Server..... | 64 |
| 3.4.2. | Instalación de Oracle Fusion Middleware® SOA Suite. | 71 |
| 3.4.3. | Instalación de Mule ESB® Community Edition | 88 |
| 3.5. | Análisis del proceso de integración de aplicaciones utilizando las herramientas escogidas para interfaces entre aplicaciones. | 90 |

| | |
|--|------------|
| 3.5.1. Integración de aplicaciones utilizando Microsoft® Biztalk® Server..... | 90 |
| 3.5.2. Integración de aplicaciones utilizando Oracle® Fusion® Middleware SOA Suite..... | 101 |
| 3.5.3. Integración de aplicaciones utilizando Mule® ESB Community Edition..... | 110 |
| 3.6. Análisis del mantenimiento futuro de las herramientas de interfaces entre aplicaciones..... | 116 |
| CAPÍTULO IV | 119 |
| 4. Caso de análisis: Integración de bases de datos financiera y de operaciones de la campaña de perforación en Petroamazonas EP..... | 119 |
| 4.1. Descripción de la situación real..... | 119 |
| 4.2. Características de las aplicaciones..... | 120 |
| 4.3. Requerimientos de integración..... | 120 |
| 4.4. Solución..... | 121 |
| CAPÍTULO V | 123 |
| 5. Conclusiones y recomendaciones..... | 123 |
| 5.1. Conclusiones..... | 123 |
| 5.2. Recomendaciones..... | 124 |
| 6. Bibliografía..... | 126 |
| 7. Glosario de términos técnicos y siglas..... | 128 |

ÍNDICE DE ILUSTRACIONES

| | |
|---|----|
| Figura A : Modelo de referencia de workflow (Hollingsworth 1995) | 3 |
| Figura B : Ciclo de vida de BPM..... | 7 |
| Figura C : Evolución de Middleware | 8 |
| Figura D : Capas externas SOA | 13 |
| Figura E : Capa de Servicios SOA | 14 |
| Figura F : Capas de la arquitectura SOA | 15 |
| Figura G : Colaboración entre servicios en coreografía | 17 |
| Figura H : Colaboración entre servicios en orquestación | 17 |
| Figura I : ESB Enterprise Service Bus..... | 18 |
| Figura J : ESB en la arquitectura orientada a servicios..... | 19 |
| Figura K : Ciclo de vida de SOA..... | 21 |
| Figura L : Modelo de madurez de SOA..... | 22 |
| Figura M : SOA y Servicios Web | 24 |
| Figura N : Estratificación del Software | 27 |
| Figura O : Integración de la capa de presentación..... | 28 |
| Figura P : Integración de la capa de datos | 29 |
| Figura Q : Integración funcional | 30 |
| Figura R : Integración Funcional por consistencia de datos..... | 32 |
| Figura S : Integración funcional por procesamiento multipaso..... | 33 |
| Figura T : Integración funcional por componentes plug & play | 34 |
| Figura U : Estructura de integración..... | 35 |
| Figura V : Biztalk Server, habilitación de IIS | 65 |
| Figura W : Biztalk Server, selección de características de SQL Server | 66 |
| Figura X : Biztalk Server, configuración de cuentas de SQL Server | 66 |
| Figura Y : Biztalk Server, instalación de paquetes adicionales s SQL Server | 67 |

| | |
|---|----|
| Figura Z : Biztalk Server, deshabilitar la memoria compartida de SQL Server | 68 |
| Figura AA : Biztalk Server, selección de componentes..... | 68 |
| Figura BB : Biztalk Server, descarga automática de características..... | 69 |
| Figura CC : Biztalk Server, finalización de la instalación | 70 |
| Figura DD : Biztalk Server, resumen de la configuración del servidor | 71 |
| Figura EE : Estructura de Oracle Fusion Middleware SOA Suite..... | 71 |
| Figura FF : Instalación de Base de datos Oracle | 72 |
| Figura GG : Instalación de Oracle Database 10g Express..... | 72 |
| Figura HH : Instalación de WebLogic Server | 73 |
| Figura II : Instalación de WebLogic Server..... | 73 |
| Figura JJ : Elección de la ruta de WebLogic Server..... | 74 |
| Figura KK : Elección de opciones de instalación de WebLogic Server | 74 |
| Figura LL : Creación de la estructura de la base de datos..... | 75 |
| Figura MM : Configuración de las variables de ambiente para la aplicación | 75 |
| Figura NN : Inicio del instalador de Repository Creation Utility..... | 76 |
| Figura OO : Ingreso de parámetros en Repository Creation Utility | 77 |
| Figura PP : Elección de las características de Repository Creation Utility..... | 77 |
| Figura QQ : Ingreso de la clave para los componentes de Repository Creation Utility..... | 78 |
| Figura RR : Creación de la estructura de Repository Creation Utility | 78 |
| Figura SS : Finalización de la creación del esquema de tablas | 79 |
| Figura TT : Instalación del servidor SOA..... | 80 |
| Figura UU : Pre Requisitos del Servidor SOA..... | 80 |
| Figura VV : Ubicación de SOA Server..... | 81 |
| Figura WW : Progreso de la instalación del servidor SOA..... | 81 |
| Figura XX : Instalación del Bus de Servicio Empresarial Oracle..... | 82 |

| | |
|--|-----|
| Figura YY : Instalación del Bus de Servicio de Oracle | 82 |
| Figura ZZ : Finalización de la instalación del Bus de Servicio de Oracle | 83 |
| Figura AAA : Instalación de JDeveloper y extensiones SOA | 83 |
| Figura BBB : Instalación de JDeveloper | 84 |
| Figura CCC : Elección de la ubicación de JDeveloper | 84 |
| Figura DDD : Opciones a instalar de JDeveloper | 85 |
| Figura EEE : Actualización de extensiones SOA | 86 |
| Figura FFF : Servidores de actualización de extensiones SOA | 86 |
| Figura GGG : Componentes a actualizar de JDeveloper | 87 |
| Figura HHH : Finalización de la instalación de Oracle Fusion Middleware SOA Suite | 87 |
| Figura III : Extracción de Mule ESB® Community Edition | 88 |
| Figura JJJ : Contenido del directorio de Mule ESB Community Edition | 89 |
| Figura KKK : Actualización de Mule ESB Community Edition | 89 |
| Figura LLL : Creación de un proyecto de Biztalk Server | 90 |
| Figura MMM : Resultado de las configuraciones de petición de estados | 94 |
| Figura NNN : Archivo public_timeline.xsd | 94 |
| Figura OOO : Estructura de un mensaje para almacenar estados de Twitter | 96 |
| Figura PPP : Creación del mapeo de datos | 96 |
| Figura QQQ : Elección de fuentes y destinos de datos | 97 |
| Figura RRR : Mapeo de datos | 98 |
| Figura SSS : Creación de la orquestación de servicios | 98 |
| Figura TTT : Creación de la estructura de mensajes para la orquestación de servicios | 99 |
| Figura UUU : Figuras de mensajes en el flujo | 100 |
| Figura VVV : Ingreso de ConstructMessage en el flujo | 100 |

| | |
|---|-----|
| Figura WWW : Estructura final del flujo de integración | 101 |
| Figura XXX : Conexión al servidor con JDeveloper | 102 |
| Figura YYY : Ingreso de credenciales de Weblogic Server..... | 102 |
| Figura ZZZ : Hostname de WebLogic | 103 |
| Figura AAAA : Creación de la infraestructura de integración | 103 |
| Figura BBBB : Elección de la tecnología de integración | 104 |
| Figura CCCC : Plantilla de integración..... | 105 |
| Figura DDDD : Plantilla de integración..... | 105 |
| Figura EEEE : Estructura de servicios Web..... | 106 |
| Figura FFFF : Propiedades del servicio web..... | 107 |
| Figura GGGG : Propiedades del servicio web | 107 |
| Figura HHHH : Propiedades de los servicios web a realizar el mapeo de datos | 108 |
| Figura IIII : selección de las funciones que actuarán sobre los datos | 109 |
| Figura JJJJ : Estructura final de integración..... | 109 |
| Figura KKKK : Conectores para Mule ESB Community Edition | 110 |
| Figura LLLL : Ubicación del conector para Twitter | 111 |
| Figura MMMM : Creación de un flujo en Mule ESB Community Edition | 111 |
| Figura NNNN : Componente HTTP de Mule ESB Community Edition..... | 112 |
| Figura OOOO : Página Twitter developers..... | 113 |
| Figura PPPP : Parámetros del conector de Twitter..... | 113 |
| Figura QQQQ : Conector Twitter en el flujo | 114 |
| Figura RRRR : Métodos disponibles del conector Twitter..... | 114 |
| Figura SSSS : Interprete de errores en el servicio | 115 |
| Figura TTTT : Ejecución del Flujo de Mule ESB para su disponibilidad..... | 116 |
| Figura UUUU : Presupuesto vs. situación real | 120 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1 : SOA vs. Otras arquitecturas..... | 11 |
| Tabla 2 : Estándares de Servicios Web | 24 |
| Tabla 3 : Microsoft® Biztalk® Server..... | 40 |
| Tabla 4 : Oracle Fusion Middleware..... | 41 |
| Tabla 5 : IBM® WebSphere Software®..... | 42 |
| Tabla 6 : SAP NetWeaver®..... | 43 |
| Tabla 7 : Informatica® PowerCenter® | 43 |
| Tabla 8 : WebMethods®..... | 44 |
| Tabla 9 : TIBCO® Business Integration®..... | 45 |
| Tabla 10 : Sonic ESB®..... | 46 |
| Tabla 11: Mule ESB Community Edition®..... | 47 |
| Tabla 12: Sopera..... | 48 |
| Tabla 13: Petals..... | 48 |
| Tabla 14: OpenEAI..... | 49 |
| Tabla 15: Apache ServiceMix..... | 50 |
| Tabla 16: Apache Camel..... | 51 |
| Tabla 17 : Cuadro comparativo de herramientas para la realización de interfaces entre aplicaciones | 55 |
| Tabla 18 : Requerimientos de Procesador de Microsoft® Biztalk® Server..... | 59 |
| Tabla 19 : Requerimiento de espacio en disco duro de Oracle Fusion Middleware SOA Suite..... | 60 |
| Tabla 20 : Requerimiento de Procesador de Oracle Fusion Middleware SOA Suite | 61 |
| Tabla 21 : Análisis de costos de las herramientas | 64 |
| Tabla 22 : Propiedades de un mensaje de petición de estados a Twitter | 93 |

| | |
|--|-----|
| Tabla 23: Estructura de almacenamiento de un estado de Twitter | 95 |
| Tabla 24 : Mantenimiento de las herramientas | 118 |

I. Resumen.

La arquitectura orientada a servicios (SOA) permite que una empresa comparta sus servicios tanto entre departamentos de la organización así como externamente entre filiales, para clientes, y empresas asociadas. Para que sea posible implementar esta arquitectura, los sistemas dentro de la empresa deben ser desarrollados de manera que se los pueda dividir en procesos, una empresa tendrá varios 'pequeños sistemas' que puedan ser invocados durante las diferentes etapas que comprendan la lógica del negocio. El acoplamiento en una arquitectura orientada a servicios es muy débil para así poder consumir y reusar cada servicio.

Con la implementación de la arquitectura orientada a servicios es necesario que estos sistemas presentados a manera de servicio se encuentren integrados para así evitar la duplicación de datos, y de tener sistemas cerrados que estos también se integren a la nueva estructura empresarial para así decidir entre una integración de datos o una integración entre la funcionalidad de los diferentes sistemas de la empresa.

Es con esta necesidad de integración que una empresa deberá implementar el Bus de Servicio Empresarial (ESB), el cual sirve como una autopista por la cual circularán las peticiones a los servicios, así como los datos que los sistemas devuelven como respuesta a un proceso, manteniendo la seguridad y la integridad de los datos.

No todos estos sistemas a integrarse en una empresa funcionan dentro de un mismo entorno, puede haber variaciones entre las bases de datos donde almacenan los resultados de los procesos, diferencia de plataformas de software donde funcionan, diferencias entre los sistemas operativos que albergan los sistemas, diferencias entre la infraestructura de hardware, por lo que tanto la arquitectura SOA como la implementación de un Bus de Servicio Empresarial deben funcionar a pesar de todas las diferencias mencionadas.

Para crear la estructura deseada de integración con la arquitectura orientada a servicios, existen múltiples herramientas de integración que abarcan los diferentes requerimientos de una empresa: tipos de licenciamiento, multiplataforma, gran

gama de bases de datos soportadas, diferentes formas de administrar los datos, trazabilidad de los procesos, seguridad de las transacciones, costo de mantenimiento, soporte, disponibilidad de documentación técnica. Es por ello que para escoger una herramienta middleware se debe tener en cuenta las múltiples variables que involucran el proceso de integración para así tomar la mejor decisión al momento de adquirir una herramienta.

II. Introducción

Toda empresa que posea procesos automatizados verá la necesidad de integrar sus diferentes sistemas de la misma manera en que las diferentes áreas integran sus flujos de trabajo y forman así una cadena de valor. Para que los sistemas se encuentren correctamente acoplados a los requerimientos del flujo de trabajo del negocio, es necesario que los sistemas usados en diferentes áreas y para diferentes propósitos se encuentren en capacidad de interactuar entre si y manejar la información que en el proceso sea generada.

La complejidad de una integración entre sistemas depende de factores como la flexibilidad de los sistemas, que tan fuerte es el acoplamiento del sistema o la metodología a usarse, por lo que la estrategia de integración debe ser claramente definida así como los procesos de la empresa deben encontrarse definidos.

Los principales sistemas corporativos (ERP¹, CRM², HR³, etc) han sido diseñados de tal forma que compartir procesos es una tarea que necesariamente involucra gran cantidad de recursos y tiempo, el enfoque de la arquitectura orientada a servicios SOA permite rediseñar la forma en que las aplicaciones presentan los resultados al usuario para crear sistemas compuestos que utilizan capacidades existentes en los nuevos sistemas, disminuyendo el impacto en las empresas por el cambio de las reglas del negocio.

La arquitectura orientada a servicios SOA separa los procesos de negocio de las funciones automatizadas organizándolas en módulos individuales dando la posibilidad de que estas puedan ser utilizadas por toda la empresa, clientes y subsidiarias. A estos módulos individuales SOA los define como servicios, *“un servicio es una funcionalidad concreta que puede ser descubierta en la red y que describe tanto lo que puede hacer como el modo de interactuar con ella”*⁴.

Al realizar una integración entre aplicaciones utilizado la arquitectura orientada a servicios SOA, se debe tener en cuenta el aplicativo que administrará las

¹ Sistemas de planificación de recursos empresariales por sus siglas en inglés (Enterprise Resource Planning)

² Sistemas de administración de clientes por sus siglas en inglés (Customer Relationship Manager)

³ Sistemas de recursos humanos por sus siglas en inglés (Human Resources)

⁴ Microsoft Corporation. La arquitectura orientada a servicios (SOA) de Microsoft aplicada al mundo real. Diciembre de 2006.

conexiones, se debe tener en cuenta la interface de integración entre 2 o más sistemas, en esta investigación se presenta al lector herramientas propietarias, así como herramientas libres para realizar la integración de sistemas utilizando la arquitectura orientada a servicios SOA.

CAPÍTULO I

1. Arquitectura orientada a servicios SOA.

1.1. *Integración de sistemas informáticos.*

Las empresas basan su operación en procesos, estos procesos generan datos para ser analizados. Los datos son importantes en toda empresa, pero no son los que la definen, ya que una empresa está definida por sus procesos y el valor que estos generan. Los procesos se caracterizan por ser largos y complejos, dinámicos, ampliamente distribuidos y personalizados, ejecutables durante períodos de tiempos variables, parcialmente automatizados y muy dependientes en la parte no automatizada de la inteligencia y del juicio de las personas.

“Los procesos, la información y las organizaciones están íntimamente relacionados”⁵, los procesos consumen, generan y transforman la información que a su vez van a estar gobernados por las políticas de la empresa. Las arquitecturas de software basado en la información ponen jerárquicamente a la información por sobre los procesos, esto genera que los procesos queden ocultos dentro de las diferentes aplicaciones usadas por la empresa durante el flujo de trabajo.

Los objetivos de una empresa se enfocan en tener flexibilidad operativa, esto para ser más competitivas frente a los cambios que pueden ocurrir en el mercado, para ello muchas veces se verán en necesidad de cambiar sus procesos, es por ello que las empresas buscan que estos cambios en los procesos se vean reflejados en los sistemas informáticos que han adquirido, es entonces cuando el enfoque de los departamentos de TI tuvo que cambiar debido a que cualquier cambio en los procesos de una empresa eran muy difíciles de plasmar en sistemas enfocados a los datos. Es esta dificultad, de plasmar los procesos de las empresas en los sistemas informáticos, que obliga a crear nuevos estándares para capturar, modelar, ejecutar y monitorizar los procesos de negocio.

A principios de los 90's aparece el interés de las empresas en dar importancia a los procesos de negocio, surgen herramientas para la realización de flujos de trabajo, tanto automatizados como manuales. Los flujos de trabajo o Workflows son definidos como: “La automatización total o parcial de los procesos de

⁵ Fischer Layna. Workflow Handbook. (2004)

negocio, que involucra el transporte de documentos, información o tareas de un participante a otro, de acuerdo a un conjunto de reglas establecidas para conseguir el objetivo global del negocio”⁶. A su vez un proceso de negocio es definido como los pasos que deben seguir –normalmente actividades- automatizadas o manuales para conseguir un objetivo concreto del negocio.

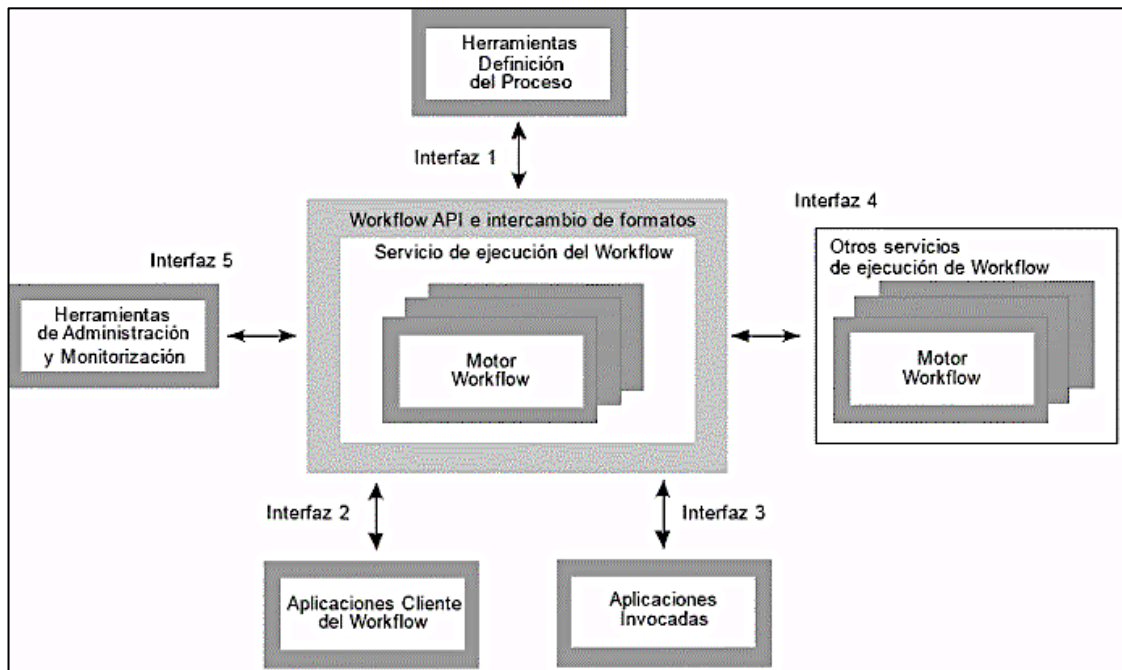
1.1.1. Sistemas de Workflow (flujo de trabajo).

Los sistemas informáticos basados en workflows poseen características comunes entre sí, es por ello que para mantener la interoperabilidad entre sistemas se ha definido los componentes comunes que deben presentar entre ellos para mantener un correcto funcionamiento, los componentes definidos son: motor de workflow, servicio de ejecución de workflow e interfaz de programación de aplicaciones de workflow.

- *Motor de workflow.* Se encarga de seguir los casos o instancias de cada proceso. El motor del flujo de trabajo provee un ambiente de ejecución para las instancias del workflow. Normalmente el motor de workflow permite interpretar las definiciones que se han realizado del proceso, controlar las instancias de un proceso tales como: creación, activación, suspensión y terminación de un proceso.
- *Servicio de ejecución de workflow.* Invoca las aplicaciones necesarias y se encarga de que las actividades sigan un orden lógico. Consiste en uno o más motores de workflow para crear, administrar y ejecutar instancias del flujo de trabajo. Existe una separación lógica entre el servicio de ejecución de workflow y el control lógico de las actividades para permitir la integración con sistemas de workflow particulares.
- *Interfaz de programación de aplicaciones de workflow.* Permite la interacción del servicio de ejecución del flujo de trabajo con otras aplicaciones.

⁶ Workflow Management Coalition. Internet. <http://www.wfmc.org/Published-Research/Article/View-category.html>

Figura A : Modelo de referencia de workflow (Hollingsworth 1995)



Fuente: Internet. <http://www.wfmc.org/Download-document/TC003v11-WfMC-Workload-Reference-Model.html>

Autor: Workflow Management Coalition

En la *Figura A*, se observa el modelo de referencia de workflow que fue desarrollado de manera genérica a partir de las aplicaciones de flujos de trabajo, en el modelo de referencia se muestra las diferentes interfaces que permitirán diferentes niveles de interacción entre los componentes de las aplicaciones. Para lograr una interoperabilidad entre aplicaciones se debe tener interfaces y el formato intercambio de datos debe ser estandarizado.

Dentro de un flujo de trabajo se encuentran diferentes categorías de datos que son manejados en los procesos. No todos los tipos de datos pueden ser manejados de la misma manera ya que hay datos más relevantes que otros, es por ello que se ha estandarizado en: Datos de control, datos relevantes y datos de aplicación.

- *Datos de control.* Datos internos que son manejados por el motor del flujo de datos y por el sistema de administración del flujo de trabajo.

- *Datos relevantes.* Estos datos sirven para determinar el estado de transición en el que se encuentra la instancia del flujo de trabajo.
- *Datos de aplicación.* Son los datos específicos de la aplicación y que no puede ser accesada por el sistema de administración del flujo de trabajo.

El modelo de workflow (*Figura A*) maneja 5 interfaces para el manejo de datos, invocación de procesos y manejo de instancias del flujo de trabajo.

- *Interfaz 1: Herramienta de definición de procesos.* En esta interfaz se permite la interacción del proceso como tal, el orden lógico que el proceso ocupa en el flujo de trabajo, los datos que intervendrán. En la interfaz de definición de procesos ocurre el intercambio de datos entre el componente que ejecuta el proceso y el servicio de ejecución del flujo de trabajo. Para el intercambio de información entre los componentes se utilizará la ayuda de metamodelos en los que se definan las entidades mínimas para definir el proceso. Para definir los metamodelos se debe usar las especificaciones para flujos de trabajo tales como XPDL⁷, BPMN⁸, BPEL⁹, YAWL¹⁰.
- *Interfaz 2: Aplicaciones clientes.* Aquí se definen las diferentes formas de acceder al motor del flujo de trabajo por medio de API's¹¹, es usado para acceder desde la aplicación de flujo de trabajo al motor del flujo de trabajo para controlar la secuencia de los procesos y actividades. También se asigna a los usuarios la carga de trabajo, las instancias que les pertenecen de cada actividad.
- *Interfaz 3: Aplicaciones invocadas.* Definición de API's para invocar las aplicaciones desde el motor del flujo de trabajo. La aplicación invocada en la interfaz 3 puede ser residente en la misma plataforma o puede estar en otra plataforma, eso sí, con una conexión de red.

⁷ Process Definition Language por sus siglas en inglés, fue diseñado para intercambiar tanto la parte gráfica como semántica de un flujo de proceso del negocio.

⁸ Business Process Model and Notation, por sus siglas en inglés, es una representación gráfica de procesos de negocio en un modelo de procesos de negocio.

⁹ Business Process Execution Language, por sus siglas en inglés, es un lenguaje ejecutable para especificar acciones entre procesos de negocio y servicios web.

¹⁰ Yet Another Workflow Language, por sus siglas en inglés, es un lenguaje de flujo de datos aceptado en software bajo licencia Open Source.

¹¹ Application Programming Interface, por sus siglas en inglés, es un código fuente diseñado para permitir la comunicación de una aplicación a otra.

- *Interfaz 4: Funciones de interoperabilidad entre sistemas de workflow.* Para alcanzar la interoperabilidad entre flujos de trabajo se define los puntos en común que comparten 2 o más procesos, por lo que puede haber 2 o más flujos de trabajo que controlen un proceso.
- *Interfaz 5: Herramientas de administración y monitoreo.* Consta de una aplicación independiente que tiene la capacidad de interactuar con los diferentes motores de flujo de trabajo y también para realizar una auditoría de los mismos. Posee las herramientas comunes para administración y monitoreo como por ejemplo: control de recursos, control de roles, estatus de los procesos, estado de los flujos de trabajo, control de transacciones.

Los sistemas de workflow han evolucionado con el paso del tiempo, es así que la *Workflow Management Coalition*¹² los describe actualmente como: “Ha pasado de ser una herramienta que manejaba únicamente el enrutado del trabajo entre la gente, hasta llegar a enrutar el trabajo de manera horizontal, es decir; entre procesos. El enrutado también puede ser vertical, controlando los pasos que van a ser llevados en cada en cada punto del flujo. Es por esta evolución que al recurso se lo puede ver como un usuario o como a un sistema completo y también puede llegar a ser una máquina. ”.

El workflow, o flujo de trabajo; se ubica en el área de integración de aplicaciones de una o varias empresas debido a que los datos se mueven entre procesos y existe una integración entre los diferentes sistemas de cada proceso y por ende el workflow es la base y guía en el enrutamiento entre aplicaciones.

1.1.2. Sistemas de administración de procesos de negocio.

Los sistemas de hoy en día se han centrado en la orientación hacia los procesos de las organizaciones, la tecnología de workflow ha presentado ciertas carencias para la integración de procesos por lo cual ha aparecido la gestión de procesos de negocio (BPM¹³). BPM define al proceso como “conjunto coordinado y completo

¹² <http://www.wfmc.org/>

¹³ Business Process Management, por sus siglas en inglés, se lo considera una evolución de los sistemas de workflow y el tratamiento automatizado de los procesos de negocio de las empresas.

de actividades colaborativas y transaccionales que proporcionan valor a los clientes”¹⁴.

BPM busca hacer los procesos de negocio más eficientes y adaptables a un ambiente más dinámico ya que las empresas promedio enfrentan retos como la gran cantidad de departamentos o la integración con socios, diferentes filiales y clientes. Pero, ¿Cómo ayuda BPM al mejor manejo de los procesos?, BPM abarca las actividades que involucran el ciclo de vida de un proceso de negocio¹⁵. El desarrollo de BPM implicó la implementación de los sistemas de administración de procesos de negocio (BPMS¹⁶) que tiene como premisa la separación de las aplicaciones de la gestión de procesos del negocio de manera que cualquier cambio en la lógica de los procesos no implique un cambio en el código fuente de las aplicaciones que intervienen en el flujo de trabajo o en su defecto, un cambio mínimo en tiempo y en costos.

BPM nos da la capacidad de identificar los servicios que una empresa necesitará de sus departamentos, nos ayuda a identificar, también, qué procesos se podrá compartir con socios y con filiales, realizar outsourcing¹⁷ de procesos de la empresa pero sin perder el control de los mismos y con la opción de monitorearlos. BPM es una metodología de mejora continua de los procesos de una empresa, por lo tanto se rige a un ciclo de vida definido y presentado a continuación:

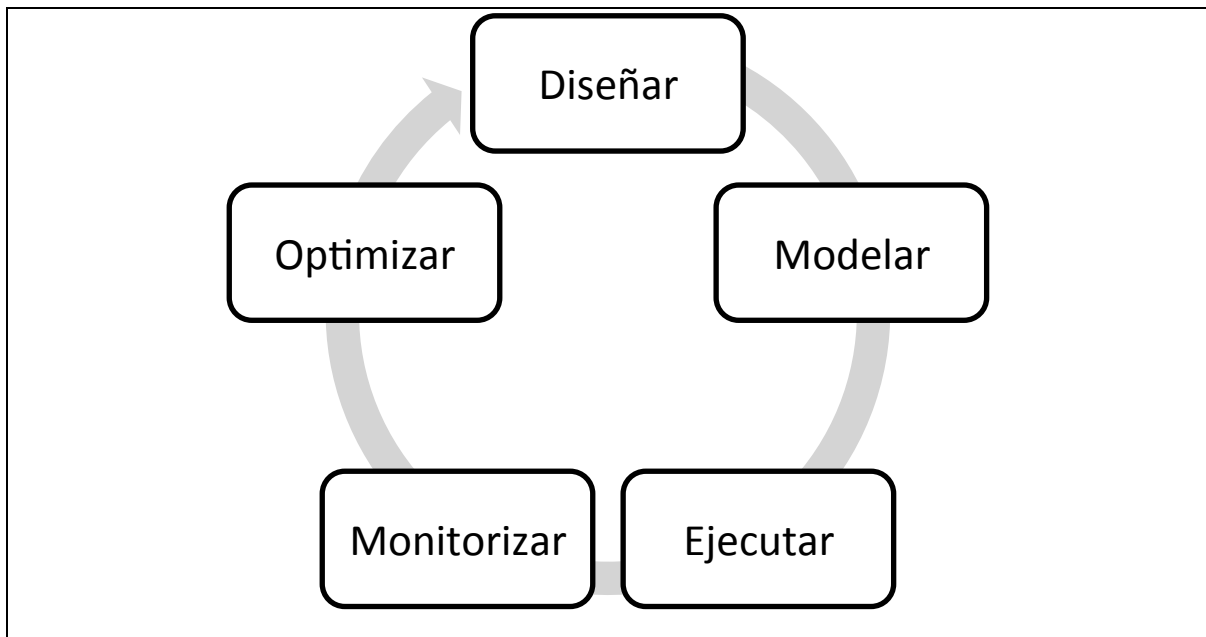
¹⁴ Smith & Fingar. Business Process Management: The Third Wave. Enero de 2004.

¹⁵ El ciclo de vida de un proceso de negocio consta de: Modelamiento, implementación, ejecución, control y gestión.

¹⁶ Business Process Management System, por sus siglas en inglés.

¹⁷ Ofertar o contratar servicios propios de un departamento de una empresa hacia otras empresas o de otras empresas.

Figura B : Ciclo de vida de BPM



Fuente: Internet. <http://www.wfmc.org/>

Autor: Juan Pablo Almeida Durán

Cuando la estructura de la empresa está correctamente definida, sus procesos de negocio han sido separados y claramente delimitados, se ha visto la necesidad de reemplazar los sistemas ERP¹⁸ que cubrían todas las áreas de la empresa por sistemas heredados y diseñados para ejecutarse de manera aislada e independiente de la plataforma, se verá en la necesidad de integrar los sistemas para así llevar a cabo la ejecución de los procesos de la empresa de una manera eficiente.

1.1.3. Estrategias de integración de sistemas.

Para lograr una integración de sistemas se tendrá la necesidad de crear una capa intermedia que sea independiente de la plataforma en que se encuentren los sistemas a integrar, a esta capa se la ha denominado Middleware y su función es proporcionar a los programadores un ambiente estándar de programación en el que no tengan que realizar cambios en el código fuente de los sistemas propietarios. Al software middleware se lo puede comparar como una tubería

¹⁸ ERP, por sus siglas en inglés, Enterprise Resource Planning, son sistemas que abarcaban todas las actividades de una empresa, lo cual hacía que sean sistemas costosos, heterogéneos y difícilmente adaptables.

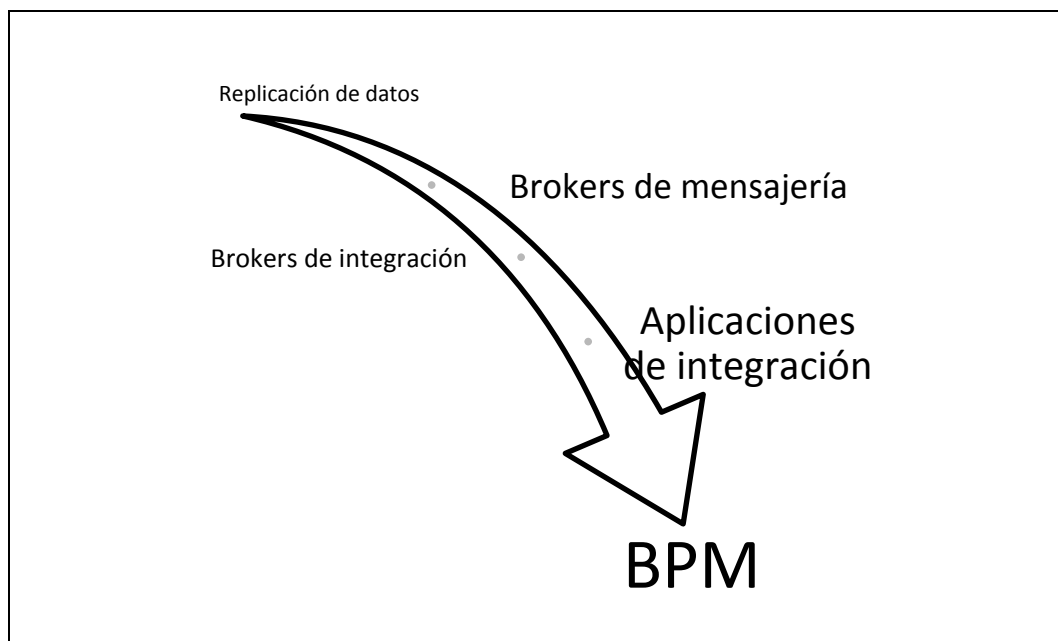
central que sirve para integrar las tuberías secundarias (aplicaciones) proporcionando un ambiente por el que circulen los datos.

Middleware está definido como: “Software de conectividad que consiste en un conjunto de servicios que permiten interactuar a múltiples procesos que se ejecutan en distintas máquinas a través de la red, ocultan la heterogeneidad y proveen de un modelo de programación conveniente para los desarrolladores de aplicaciones”¹⁹.

El software Middleware ha ido evolucionando con el paso del tiempo creando múltiples soluciones para integrar aplicaciones, desde herramientas que únicamente replican datos de un sistema hacia otro, intermediarios de mensajes, herramientas de flujos de datos, arquitecturas orientadas a servicios, hasta BPM.

En la Figura C, se encuentra la evolución de Middleware a través del tiempo y los tipos de herramientas que han sido heredados de la investigación de estos sistemas.

Figura C : Evolución de Middleware



Fuente: Internet. <http://www.wfmc.org/>

Autor: Juan Pablo Almeida Durán

¹⁹ IETF, Internet Engineering Task Force - 2005

Los intermediarios de mensajería y los sistemas basados en workflow son idóneos para integración de aplicaciones dentro de la misma organización, por ejemplo entre departamentos, con la aparición de internet las integraciones se las puede realizar a una escala mucho más grande proporcionando protocolos estándar así como los formatos de datos, es así como aparecen los servicios web que son aplicaciones capaces de interactuar en la web y también son capaces de intercambiar datos entre sí, dando y recibiendo servicios.

La aparición de los servicios web contribuyó a solucionar problemas que se presentaban en tecnologías middleware anteriores, ello gracias a: Arquitecturas orientadas a servicios, Protocolos, Estandarización.

- *Arquitecturas orientadas a servicios.* La funcionalidad de los sistemas se divide en servicios que estos presentan, el acoplamiento de los sistemas es débil permitiendo que se dividan en módulos, y así facilitar integraciones.
- *Protocolos.* Los protocolos middleware fueron rediseñados permitiendo que las conexiones entre empresas se realicen sin intermediarios, los procesos ya no son controlados por un solo sistema centralizado.
- *Estandarización.* A pesar de que consorcios de estandarización han logrado estandarizar aspectos como contenido de los mensajes entre aplicaciones, el lenguaje utilizado y los protocolos de integración, todavía existe software propietario que utiliza su propia estandarización haciendo que una posible integración sea un proceso más largo y costoso.

1.1.4. Arquitectura basada en servicios web.

Una arquitectura basada en servicios web consta de 3 elementos principales, el proveedor del servicio web, el registro del servicio web y el consumidor del servicio web. El proveedor del servicio abarca la plataforma en donde se encuentra el servicio o aplicación. El consumidor del servicio es la aplicación que invoca el servicio web que puede ser invocado por un usuario o por otro servicio web. El registro del servicio web consiste en un registro donde buscar los servicios y sus descripciones.

Un paso importante para la integración de aplicaciones es que la empresa defina la arquitectura a utilizar, según sus requerimientos, debe analizar la capacidad de integrar las aplicaciones heredadas ya existentes con las aplicaciones nuevas que se vayan incorporando. Para que una empresa no se encuentre vulnerable debido a posibles cambios en sus procesos se debe buscar una arquitectura flexible a dichos cambios con herramientas para simplificar eventuales modificaciones, el diseño y la implementación de los procesos, y diseñar sus aplicaciones en forma de servicios, de esta manera, permitir que un proceso se fragmente de acuerdo a los servicios que lo conforman.

Utilizando el enfoque mencionado la empresa tendrá la capacidad de hacer que sus procesos sean fácilmente acoplables a procesos de diferentes empresas, prestar servicios de sus procesos y reestructurar sus procesos sin tener que realizar cambios mayores en su arquitectura de software, esta arquitectura que se orienta a los servicios se denomina SOA (Service Oriented Architecture).

1.2. Introducción a la arquitectura orientada a servicios SOA.

La orientación a servicios nace de la orientación a objetos y su evolución hacia la orientación a componentes, la arquitectura orientada a servicios propone convertir el software en servicios disponibles, con un acoplamiento débil, fáciles de integrar y fáciles de reutilizar. SOA establece una arquitectura diseñada para la integración de aplicaciones independientes de manera que las aplicaciones puedan ser accedidas a través de la red. La forma habitual de implementar SOA es a través de servicios web, ya que se basan en estándares y son independientes de la plataforma. SOA puede descomponer aplicaciones monolíticas en un conjunto de servicios para implementarlos de forma modular.

SOA es iterativa, ya que un servicio puede estar compuesto de otros servicios de menor jerarquía pudiendo ser invocados basándose en decisiones tomadas de acuerdo a la lógica del proceso de negocio, con esto un desarrollador que decida ocupar un servicio ya diseñado anteriormente, no debe preocuparse de la composición interna del servicio sino de los parámetros de entrada y de salida, resultado del funcionamiento interno del servicio. Cada servicio deberá ser desarrollado de manera que aporte valor directo a la lógica del proceso del negocio para así no tener un sinnúmero de servicios que en lugar de proporcionar

valor a la empresa, se transformen en un problema de manejo para la infraestructura de IT.

Para comprender SOA es inevitable realizar una comparación con las diferentes arquitecturas de programación que han ido evolucionando a través del tiempo, así se tiene a la Programación estructurada, Programación orientada a objetos, Programación orientada a componentes y al final la orientación a servicios que nos proporciona la arquitectura SOA.

Tabla 1 : SOA vs. Otras arquitecturas

| | Programación estructurada | Objetos | Componentes | Servicios (SOA) |
|----------------------------------|----------------------------------|--------------------|--------------------|------------------------|
| Granularidad²⁰ | Muy fina | Fina | Intermedia | Gruesa/Fina |
| Reusabilidad | Baja | Baja | Intermedia | Alta |
| Acoplamiento | Fuerte | Fuerte | Débil | Muy Débil |
| Ámbito de comunicación | Intra - aplicación | Intra - aplicación | Inter - aplicación | Inter- Empresas |

Autor: Juan Pablo Almeida Durán

Al referir que un sistema SOA presenta una granularidad gruesa o fina, se habla de que sus servicios (aplicaciones) se encuentran estructurados de manera tal que su complejidad permite una fácil modificación. Normalmente el tamaño de un sistema termina influyendo en su granularidad, cuando un sistema posee servicios complejos y que realizan grandes operaciones la tendencia será a que la granularidad sea fina y por ende la funcionalidad del servicio será de mayor calidad, la granularidad la definirán los desarrolladores de los servicios.

En cuanto a la reusabilidad que nos proporciona SOA, es alta ya que al separar los componentes del sistema en servicios se pueden ocupar los diferentes módulos en las diferentes capas de la arquitectura, por ejemplo; en un sistema de inventario se puede ocupar un servicio que devuelva el stock de cierto producto

²⁰ Es el grado en el que un sistema se puede descomponer en partes. A mayor granularidad mayor capacidad de reutilización.

en diferentes partes del proceso, tanto como para la parte financiera como para la parte operativa.

La dependencia que existe entre los diferentes elementos de SOA es muy débil (acoplamiento) permitiendo así que cualquier cambio en la lógica de un servicio tenga un impacto mínimo o nulo en los demás servicios utilizados en el sistema, para ello la estrategia de implementación de los servicios tiene que tener un alto nivel de abstracción de la lógica del negocio y no caer en la realización de servicios redundantes o en la realización de servicios que podrían formar parte de un solo servicio.

1.2.1. Formas de implementar SOA en la empresa.

Existen 2 formas de implementación de SOA en una empresa: Top Down (descendente) y Bottom Up (ascendente), que se diferencian en la manera de abordar el problema.

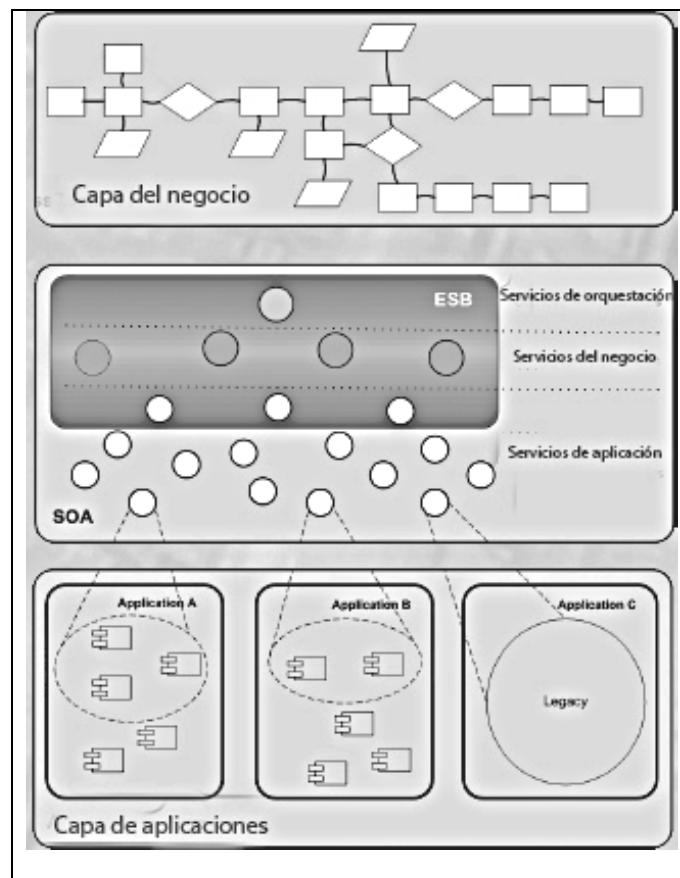
- La metodología Top Down (descendente), es ideal cuando se implementa un nuevo sistema ya que esta metodología supone analizar todos los procesos de negocio de la empresa, lo cual supone tiempos extensos de implementación así como una gran inversión. La ventaja de esta metodología es que se tendrá un conocimiento amplio de las reglas del negocio, las dependencias se identificarán de manera temprana pudiéndose realizar los servicios de manera más certera, se tendrá un conocimiento de la complejidad del intercambio de mensajes con subsidiarias o socios de la empresa de manera temprana.
- La metodología Bottom Up (ascendente), se la utiliza cuando se desea realizar migraciones de arquitectura, cuando se desea implementar SOA en una empresa ya constituida, con sistemas ya implementados, el costo inicial de implementación es bajo ya que se va introduciendo gradualmente SOA de acuerdo a los requerimientos que vayan apareciendo. Con esta metodología no va a ser mandatorio el conocimiento de los procesos del negocio, sino conocer el funcionamiento de los sistemas.
- Microsoft introdujo una metodología híbrida denominada Middle Out (centro afuera) que toma algunas partes de las 2 metodologías anteriores, pero toma las partes esenciales de la visión del negocio para establecer

prioridades en el proyecto y luego realizar un proceso iterativo para crear pequeños fragmentos de funcionalidades y así ir obteniendo un retorno de la inversión en un plazo más corto.

1.2.2. Estructura de SOA.

El paradigma de la arquitectura orientada a servicios posee 3 capas externas: La capa de negocio, capa de servicios y capa de aplicaciones. Las capas numeradas vienen a ser una visión abstracta de toda la visión empresarial de SOA, más adelante en esta disertación se indicará propiamente las capas que conforman SOA.

Figura D : Capas externas SOA



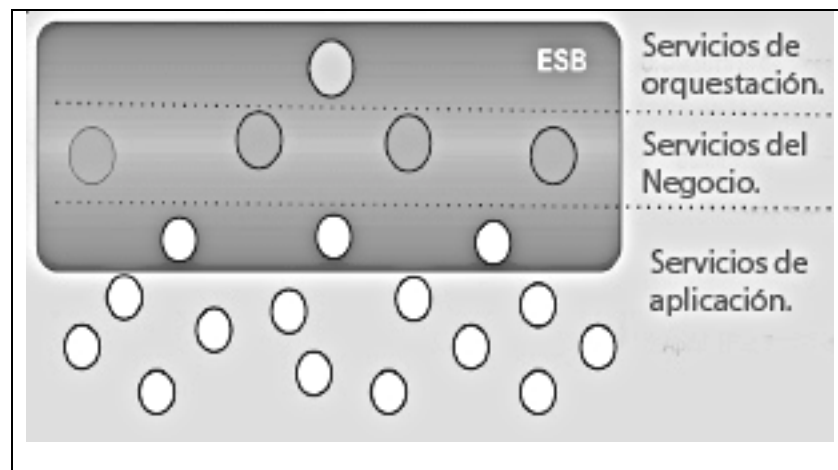
Fuente: Internet. <http://www.theserverlabs.com/soa-quickstart-solution.html>

La capa del negocio deberá manejar toda la lógica del mismo en forma de servicios, el lenguaje idóneo para manejar la lógica del negocio es el Lenguaje de Ejecución de Procesos de Negocio (BPEL), en esta capa se puede realizar todos los cambios en cuanto a procesos internos del negocio se refiere.

La capa de aplicaciones engloba el software propietario y personalizado que es utilizado en los diferentes procesos de la empresa, se debe identificar a los programas que intervienen directamente en la lógica del negocio para que estos sean publicados en el bus de datos empresarial.

Bajo la arquitectura SOA es indispensable crear una capa de servicios la cual funcione independientemente de la capa de aplicaciones para así tener independencia de la plataforma en la que se encuentran las aplicaciones, en la capa de servicios se encuentra los siguientes componentes: servicios de aplicación, servicios del negocio y servicios de orquestación

Figura E : Capa de Servicios SOA



Fuente: Internet. <http://www.theserverlabs.com/soa-quickstart-solution.html>

- Los servicios de aplicación constituyen la parte de más bajo nivel dentro de la capa de servicios, son la base para crear servicios del negocio, proporcionan soluciones independientes de los procesos del negocio, acá van las aplicaciones que la empresa ha adquirido y que comúnmente son sistemas heredados (wrapper²¹), estos servicios de aplicación proporcionan acceso a las fuentes de datos, los servicios de aplicación poseen diferentes niveles de abstracción, siendo únicamente los servicios ligados al negocio los que serán expuestos en el bus de datos empresarial (ESB²²).

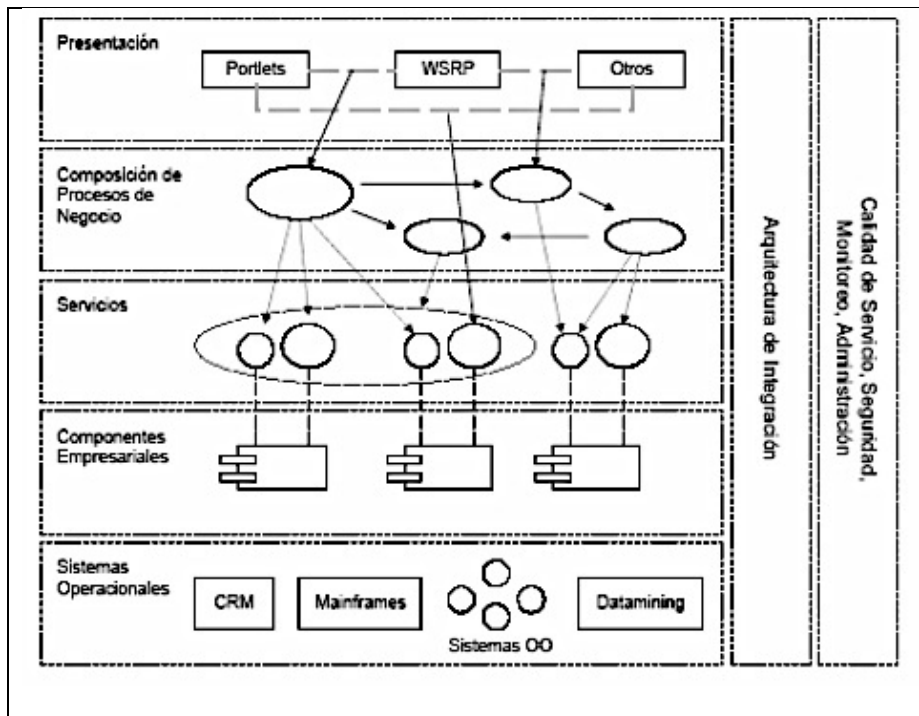
²¹ Grupo de subrutinas o clases utilizadas para desarrollar software y que se encuentran empaquetadas mostrando únicamente su funcionalidad.

²² Modelo para diseñar e implementar la interacción entre aplicaciones de software en la arquitectura orientada a servicios.

- Los servicios del negocio expresan la lógica empresarial, estos pueden encapsular en un servicio, la lógica de una tarea inherente a uno o varios procesos empresariales, estos procesos al estar involucrados directamente con la lógica de la empresa deben ser incluidos en el bus de datos empresarial.
- Los servicios de orquestación están compuestos por los servicios de aplicación y los servicios del negocio, la manera en que se componen los servicios depende de las reglas del negocio y la secuencia de ejecución, estos servicios también son incluidos en el bus de datos empresarial.

Profundizando en las capas que posee la arquitectura SOA de acuerdo a la tecnología se tiene 7 capas ²³ que son un subconjunto de las capas mencionadas anteriormente.

Figura F : Capas de la arquitectura SOA



Autor: Arsanjani, Borges, Holley. 2004

Fuente: "Service-oriented architecture: Components and modeling can make the difference"

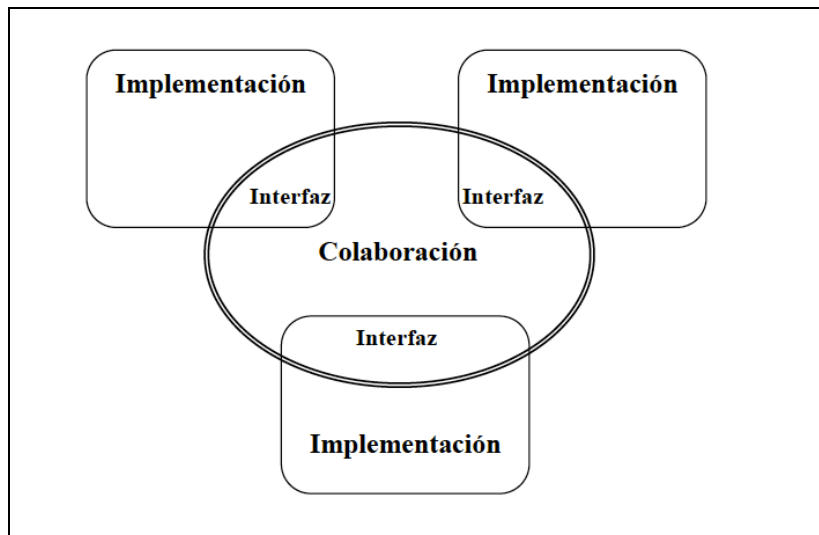
²³ Arsanjani, Borges, Holley. 2004. "Service-oriented architecture: Components and modeling can make the difference."

- *Capa 1: Sistemas Operacionales.* Esta capa contiene aplicaciones y sistemas ya existentes, sistemas heredados, SOA tiene la capacidad de reutilizar estos sistemas e integrarlos.
- *Capa 2: Componentes Empresariales.* Estos componentes realizan la funcionalidad y mantenimiento de la calidad del servicio que se brinda.
- *Capa 3: Servicios.* En esta capa se encuentran los servicios que la empresa ha decidido compartir, pueden ser descubiertos, referenciados directamente, o ser parte de una orquestación o de un servicio compuesto. Normalmente estos servicios exponen la funcionalidad de negocio a través de procesos de negocio o contratos que permiten invocar los componentes de negocio que se encuentran en la capa de componentes de la empresa. Estos contratos permiten cambiar la forma en que se llevan a cabo las tareas sin necesidad de modificar los servicios expuestos.
- *Capa 4: Composición de Procesos del Negocio.* Define la estructura de los procesos del negocio teniendo en cuenta los servicios presentados en la capa 3, los servicios se introducen a través de un flujo de orquestación.
- *Capa 5: Presentación o Acceso.* Esta capa se encuentra normalmente fuera del ámbito SOA ya que SOA desacopla la interface de usuario de los componentes, pero la nueva tendencia tecnológica y la orientación a Web 2.0 está dando mayor importancia a los servicios en la capa de presentación.
- *Capa 6: Integración de Servicios.* Esta capa también es denominada ESB por sus siglas en inglés y es el bus de servicios de la empresa, en esta capa se tiene la capacidad de integrar los diferentes servicios a través de enrutamiento inteligente, para ello se utilizará software de interfaces de aplicaciones.
- *Capa 7: Calidad del Servicio.* Esta capa permite monitorear, mantener la calidad del servicio, proporcionar seguridad al sistema. Se compone de software especializado en monitorear las aplicaciones SOA y, de haberlos; los servicios web.

Una práctica común en la arquitectura SOA es la coreografía y orquestación de servicios, la coreografía se refiere al intercambio de mensajes entre actividades

desarrolladas en un proceso de negocio, independientemente del comportamiento interno de los mismos, la petición de una acción a un servicio con su respectiva confirmación de ejecución es la coreografía entre quien va a consumir el servicio y el servicio a ser consumido.

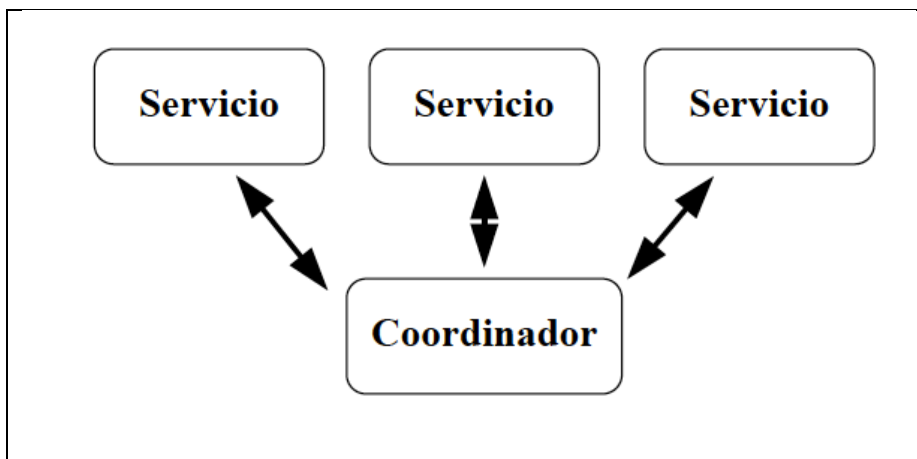
Figura G : Colaboración entre servicios en coreografía



Fuente: Internet. <http://www.01.ign.es/ign/resources/acercaDe/aig/B.pdf>

La orquestación de servicios se refiere a un proceso de negocio que al ser ejecutado permite interactuar con servicios web tanto internos como externos. En el caso de la orquestación una de las partes involucradas siempre tendrá el control de la actividad. Para lograr la orquestación de servicios es necesario que la lógica del negocio se encuentre separada del servicio web invocado, esto se logrará con la ayuda de un motor de orquestación.

Figura H : Colaboración entre servicios en orquestación



Fuente: Internet. <http://www.01.ign.es/ign/resources/acercaDe/aig/B.pdf>

Los motores de orquestación en una arquitectura SOA permiten a los procesos conectarse entre si, ya que manejan toda la lógica del negocio (a manera de flujos de trabajo). El motor de orquestación más utilizado y que ya es considerado como un estándar es BPEL.

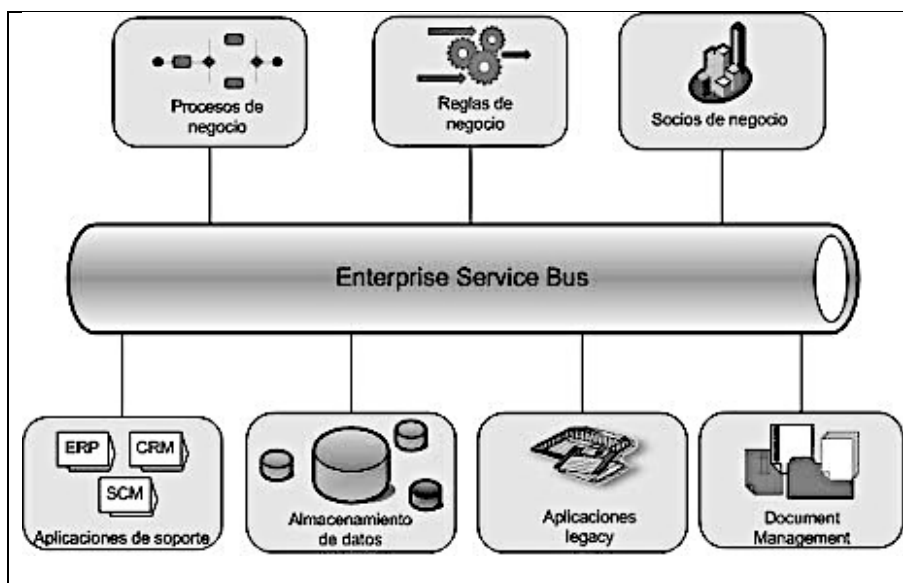
1.3. Principios para el desarrollo, implementación y mantenimiento de SOA.

Cuando se habla de una arquitectura orientada a servicios (SOA), necesariamente se debe mencionar el elemento neurálgico que habilite esta arquitectura en una corporación y así poder realizar la interacción de sus servicios dentro de la misma empresa o con otras empresas, este elemento es conocido como Bus de Servicio Empresarial (ESB²⁴).

1.3.1. Bus de servicio empresarial (ESB).

Un ESB es una infraestructura de transporte de los mensajes entre el motor de procesos y los servicios que posee la empresa. ESB es un orquestador de servicios que permite interactuar de mejor forma a los componentes técnicos y de información de una empresa con los componentes del negocio (reglas del negocio, flujos de trabajo).

Figura I : ESB Enterprise Service Bus

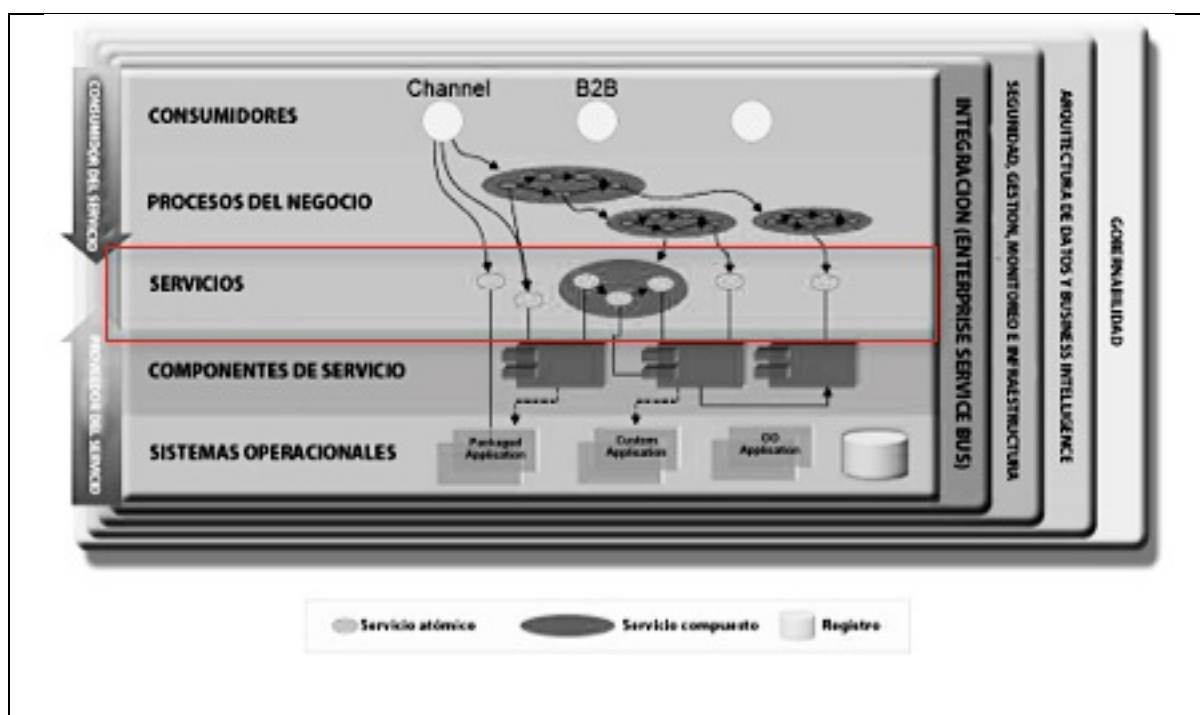


Fuente: Internet. <http://kaf.com.mx/home/font-styles-mainmenu-54>

²⁴ Enterprise Service Bus por sus siglas en inglés.

“ESB permite a la empresa conectar, mediar y controlar la interacción entre diversas aplicaciones y servicios a lo largo de entornos altamente distribuidos y heterogéneos”²⁵. Como se puede ver en la Figura J, en el recuadro rojo se observa la utilidad de ESB dentro de SOA, siendo el vínculo entre el consumidor de un servicio y el proveedor del mismo. El Bus de Servicio empresarial es considerado el verdadero habilitador de una arquitectura orientada a servicios ya que proporciona la infraestructura para la integración de servicios independientemente de su sistema operativo.

Figura J : ESB en la arquitectura orientada a servicios



Fuente: Internet. <http://technology.amis.nl/2006/05/oracle-enterprise-service-bus-esb-a-first-impression-and-its-not-bad-at-all/>

Autor: Ing. Enrique Chaviano Gómez

“El Bus de Servicios Empresariales (ESB) es un backbone²⁶ de integración basado en estándares, que combina capacidades de mensajería, servicios web, transformaciones y ruteo inteligente”²⁷ y sus objetivos son:

²⁵ Elementos de SOA. Ing. Enrique Chaviano Gómez. (15-05-2009) <http://www.gestiopolis.com/administracion-estrategia/erp-arquitectura-orientada-a-servicios.htm>

²⁶ Backbone Es una infraestructura diseñada para el transporte de datos en una red.

²⁷ Elementos de SOA. Ing. Enrique Chaviano Gómez. (15-05-2009)

- Proveer una herramienta única de integración con independencia de sistemas operativos, protocolos, lenguajes de programación, topologías de red y repositorios de datos.
- Minimizar el tiempo de desarrollo de nuevos servicios.
- Monitoreo de los servicios y manejo de la seguridad de los datos transportados.

1.3.2. Estructura del bus de servicios empresariales (ESB).

En un bus de servicios empresariales se encuentran 3 capas primarias:

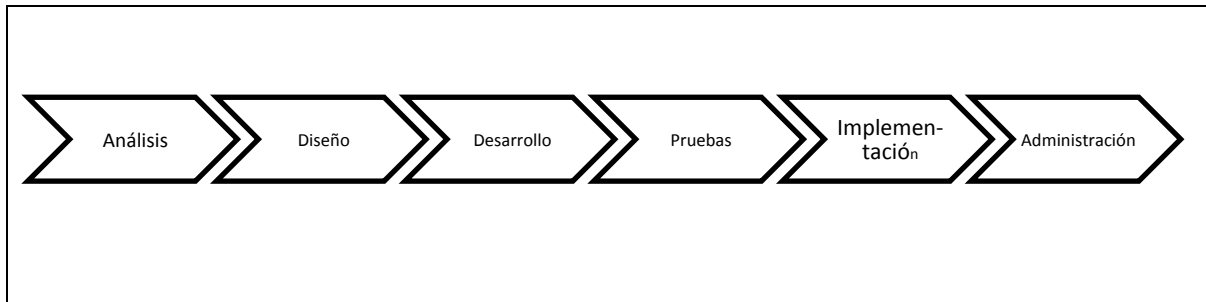
- Capa de presentación (Front End). Es la interfaz mediante la cual el usuario interactúa con la solución a esta capa pertenecen los sistemas externos, aplicaciones departamentales.
- Capa de integración (Middleware). Esta capa se encarga de presentar los servicios a ser consumidos por la capa de presentación, en esta capa se realiza la integración de aplicaciones por su capacidad de transformación y ruteo de los datos.
- Capa de datos (Back End). Esta capa provee los datos requeridos por la capa de integración para cumplir los requerimientos de la capa de presentación. En esta capa se encuentra las bases de datos, sistemas heredados llamados "Legacy"²⁸.
- Los conectores son piezas de software que sirven de nexo entre el Middleware con la capa de datos y capa de presentación, de acuerdo a su función pueden ser clientes (Middleware -> Capa de presentación) o servidores (Middleware -> Capa de datos).

²⁸ Sistemas que han quedado obsoletos pero que siguen siendo utilizados por la empresa.

1.4. Ciclo de vida y modelo de madurez de la arquitectura orientada a servicios (SOA).

1.4.1. Ciclo de vida de SOA

Figura K : Ciclo de vida de SOA



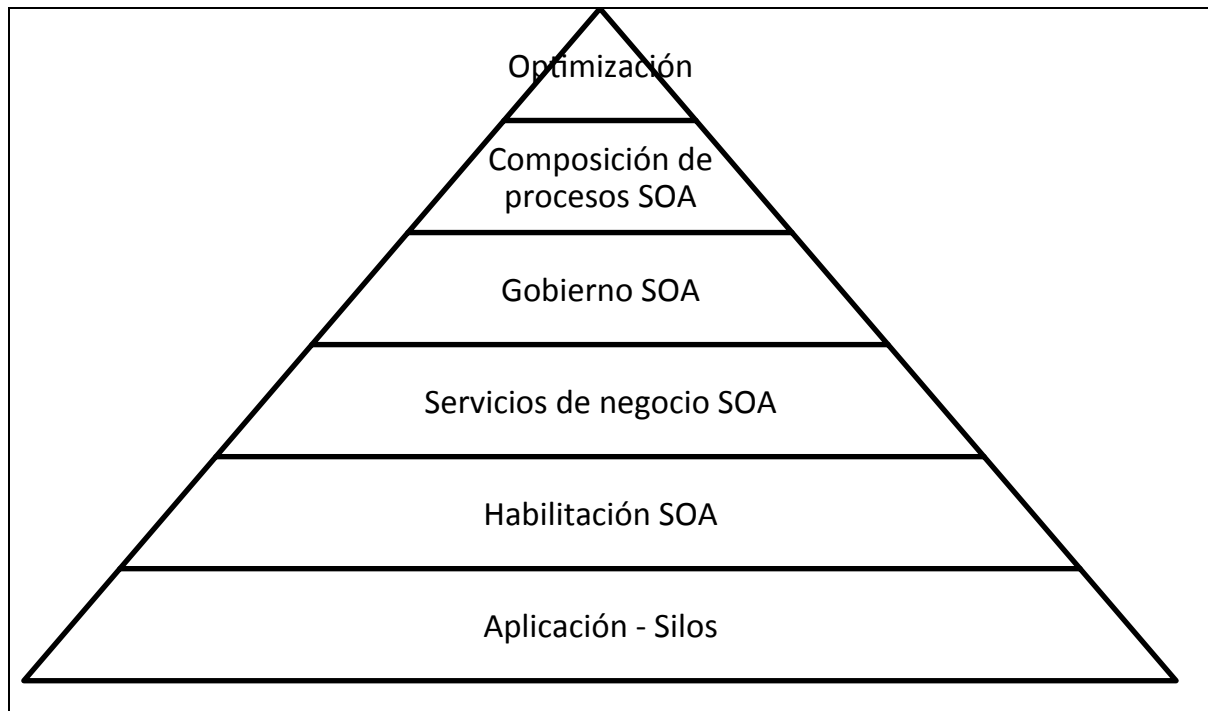
Fuente: Internet. <http://blogs.vass.es/?p=423>

Autor: Consulting IT service outsourcing

- Análisis. En esta etapa se debe definir, a partir de los flujos y reglas del negocio, los procesos que deben ser expuestos como servicios.
- Diseño. En el diseño de la solución se deberá definir los estándares a utilizar incorporando convenciones y principios para sistemas orientados a servicios.
- Desarrollo. En la etapa de desarrollo se debe construir la arquitectura de acuerdo a los parámetros establecidos.
- Pruebas. Las pruebas serán realizadas sobre los servicios implementados, el acoplamiento de los mismos y la mensajería entre aplicaciones.
- Implementación. En la implementación se deberá utilizar las herramientas definidas como middleware y poner en marcha la solución.
- Administración. En la administración de una solución SOA se deberá monitorear los servicios, los mensajes entre aplicaciones y los posibles cuellos de botella.

1.4.2. Modelo de madurez de SOA.

Figura L : Modelo de madurez de SOA



Fuente: Internet.

http://www.amis.org.mx/InformaWeb/Documentos/Archivos/4_Gabriel%20Lopez%20_Software%20AG.pdf

Autor: René Monroy – Gabriel Vásquez

- Nivel 0. (Aplicación). No se encuentra implementado SOA, las aplicaciones se manejan separadamente sin integración ni enfoque a servicios.
- Nivel 1. (Habilitación SOA). En esta etapa se realiza el enfoque de los procesos del negocio a manera de servicios a ser consumidos por el usuario de la aplicación. Se modernizan los sistemas existentes.
- Nivel 2. (Servicios de negocio SOA). Se define el crecimiento de los servicios del negocio. Dentro de este nivel también están los servicios colaborativos, que son aquellos que se compartirán para ser consumidos por los diferentes departamentos de la organización o con otras empresas.
- Nivel 3. (Gobierno SOA). Se consolidan los procesos de negocio en forma de servicios.
- Nivel 4. (Composición de procesos SOA). Se analizan los resultados de los servicios.

- Nivel 5. (Optimización). En este nivel se analizan los servicios para encontrar puntos de mejora continua.

1.5. Estándares manejados para la implementación de SOA.

La arquitectura orientada a servicios se puede implementar en una empresa utilizando una gran gama de tecnologías como por ejemplo: CORBA²⁹, DCOM³⁰, Servicios Web, POJO³¹, EJB³², MOM³³, etc. Ya que estas tecnologías cumplen con la independencia de plataforma, bajo acoplamiento y autonomía se puede realizar una implementación de SOA con las mismas. La experiencia en implementaciones de la arquitectura orientada a servicios muestran que la implementación con servicios web es óptima y se puede hablar del paradigma SOA con implementación mediante servicios web.

El problema con tecnologías como CORBA, DCOM, POJO, EJB, etc. es que a pesar de cumplir con los requerimientos de SOA de una manera teórica, no la cumplen de una manera práctica ya que muchas de estas tecnologías fueron diseñadas para definir nuevos estándares en lugar de utilizar los ya existentes, un ejemplo de esto es CORBA, cuya implementación requiere de la utilización de un protocolo exclusivo (IIOP³⁴) y no como la implementación por servicios web que utiliza protocolos existentes (HTTP, HTTPS) al igual que esquemas de datos definidos (XML).

Una implementación SOA de un sistema mediante Servicios Web maneja directrices y estándares variados para el transporte de mensajes, exposición de servicios web, exposición en la red.

²⁹ Common Object Request Broker Architecture.

³⁰ Distributed Component Object Model.

³¹ Plain Old Java Objects.

³² Enterprise Java Bean.

³³ Message Oriented Middleware.

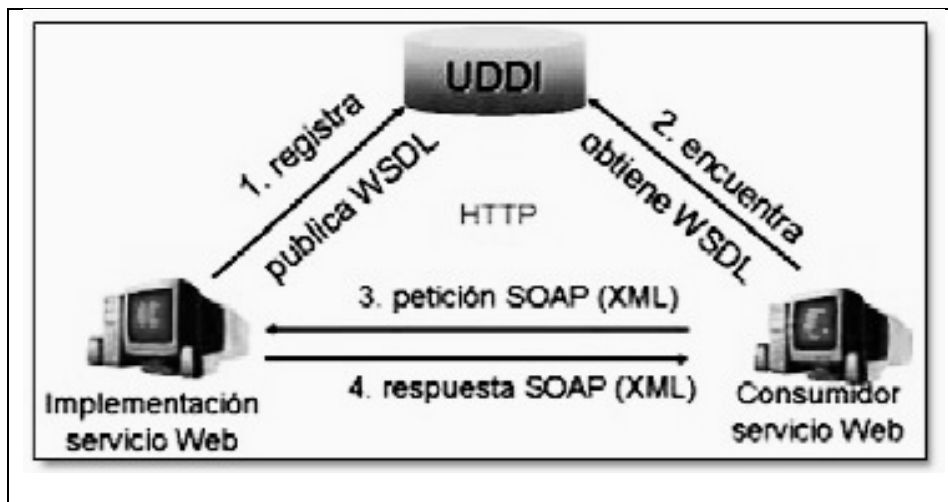
³⁴ Internet Inter-ORB protocol, Protocolo que funciona bajo TCP/IP

Tabla 2 : Estándares de Servicios Web

| Estándares de servicios web (SOA) | |
|--|--|
| XML ³⁵ | Intercambio de información estructurada entre diferentes plataformas. |
| WSDL ³⁶ | Describe las necesidades formales de un servicio web. |
| UDDI ³⁷ | Es utilizado en servicios web para que exista un registro del mensaje SOAP en la arquitectura. |
| SOAP ³⁸ | Define la comunicación entre procesos en entornos distribuidos. |

En la Figura M, se observa la forma en que los servicios web interactúan en una arquitectura orientada a servicios y donde son manejados los estándares mencionados en la Tabla 2.

Figura M : SOA y Servicios Web



Fuente: Internet.

[http://www.di.uniovi.es/~labra/cursos/Web20/ServiciosWeb.html#\(1\)](http://www.di.uniovi.es/~labra/cursos/Web20/ServiciosWeb.html#(1))

³⁵ Extensible Markup Language.

³⁶ Web Services Description Language.

³⁷ Universal Description Discovery and Integration.

³⁸ Simple Object Access Protocol.

Otro estándar utilizado en una SOA es el que permite la invocación de un servicio desde un proceso de negocio, WS-BPEL³⁹, la necesidad de desarrollar este estándar fue que al poseer servicios creados en diferentes lenguajes de programación, debía existir una forma estandarizada de invocarlos al ser necesitados en un proceso de negocio. El lenguaje BPEL es de orquestación diseñado para el control centralizado de invocación de servicios web, aplicando la lógica del negocio.

³⁹ Web Services Business Process Execution Language

CAPÍTULO II

2. Herramientas de integración de sistemas, compatibles con SOA.

Existen diversas problemáticas que se dan en las fases previas, durante el desarrollo y posteriores a realizar una integración de aplicaciones. En la fase previa a realizar una integración de aplicaciones se encuentran con sistemas aislados, tradicionalmente conocidos como silos, ya que presentan una verticalidad en su funcionalidad cuando la tendencia es que las soluciones de software presenten horizontalidad, es decir; que intervengan en múltiples procesos.

Normalmente una empresa decide realizar una integración de aplicaciones al tener la necesidad de soportar procesos que involucran varias herramientas de software y al tener información inconsistente en distintos sistemas. En las fases posteriores a una integración de datos se debe evitar, en lo posible, una dependencia entre sistemas o que existan integraciones aisladas sin usar patrones unificados.

De acuerdo al estilo de integración que sea necesario realizar en la empresa existen herramientas middleware⁴⁰ especializadas para cada diferente tipo de integración, en esta investigación se enfoca en las herramientas que abarcan todos los tipos de integración, sin necesidad de usar diferentes licencias de software.

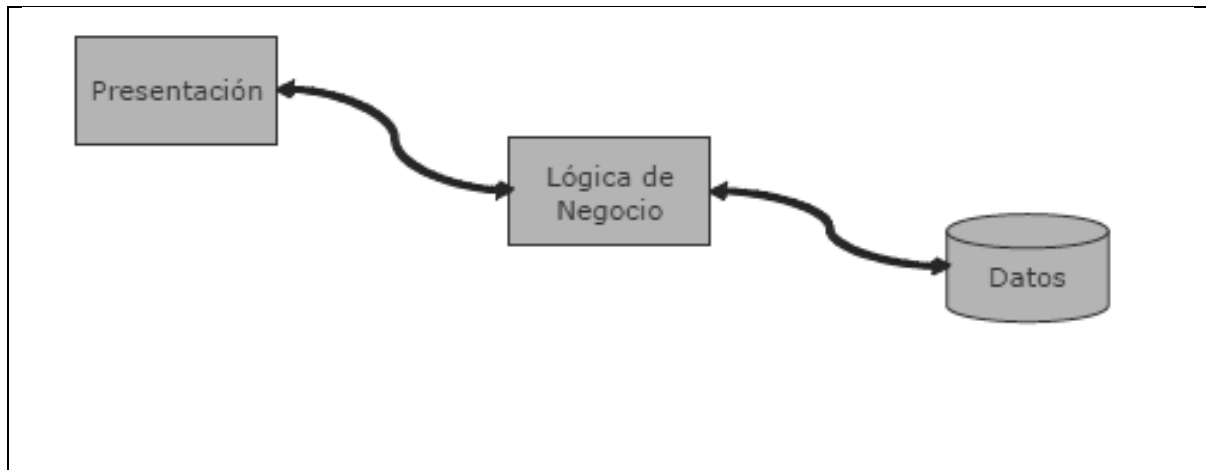
2.1. Descripción de la funcionalidad de una interfaz entre aplicaciones.

Para hablar de una interfaz entre aplicaciones, se debe abordar la estratificación del software que consiste en la división del software en capas para reducir la complejidad, es así que en las aplicaciones típicas se tienen tres capas: Presentación, Lógica del Negocio y Datos.

⁴⁰ Interface de unión entre componentes de software

- Capa de presentación. Es la capa de interacción con el usuario.
- Capa de lógica del negocio. Es la funcionalidad que el sistema debe realizar de acuerdo al entorno de trabajo para el que fue programado.
- Capa de datos. Es la lógica para la comunicación con otros sistemas que realizan tareas para la aplicación.

Figura N : Estratificación del Software



Autor: Juan Pablo Almeida Durán

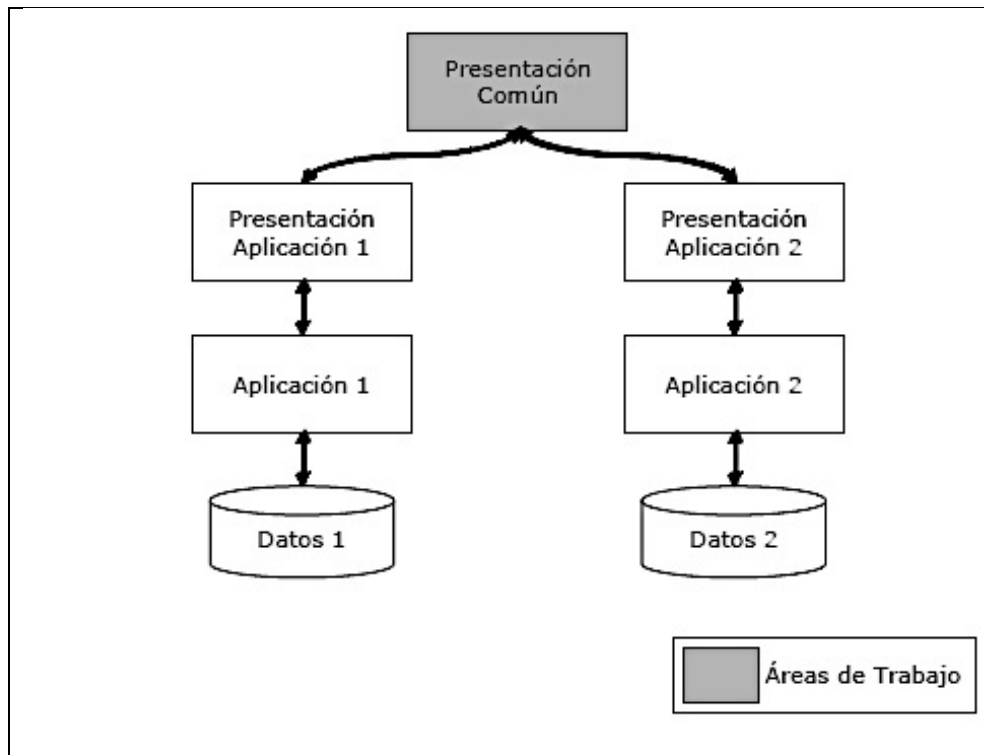
Una interfaz entre aplicaciones proporciona las condiciones ideales para permitir la integración de aplicaciones en cualquiera de sus capas, pero con ciertas particularidades que se ven en el desarrollo de este capítulo.

2.1.1. Integración de la capa de presentación.

Este es el modelo de integración que posee más limitaciones, sobre todo de rendimiento. Este modelo de integración se lo realiza cuando las aplicaciones originales limitan el acceso a la capa de datos de las mismas, este modelo de aplicación consiste en desarrollar una interfaz de usuario que unifique varias aplicaciones.

En este modelo de aplicación no se modificará la capa de la lógica del negocio y tampoco se tendrá acceso a la capa de datos, es decir las aplicaciones funcionarán como silos pero poseerán una interfaz de usuario común.

Figura O : Integración de la capa de presentación

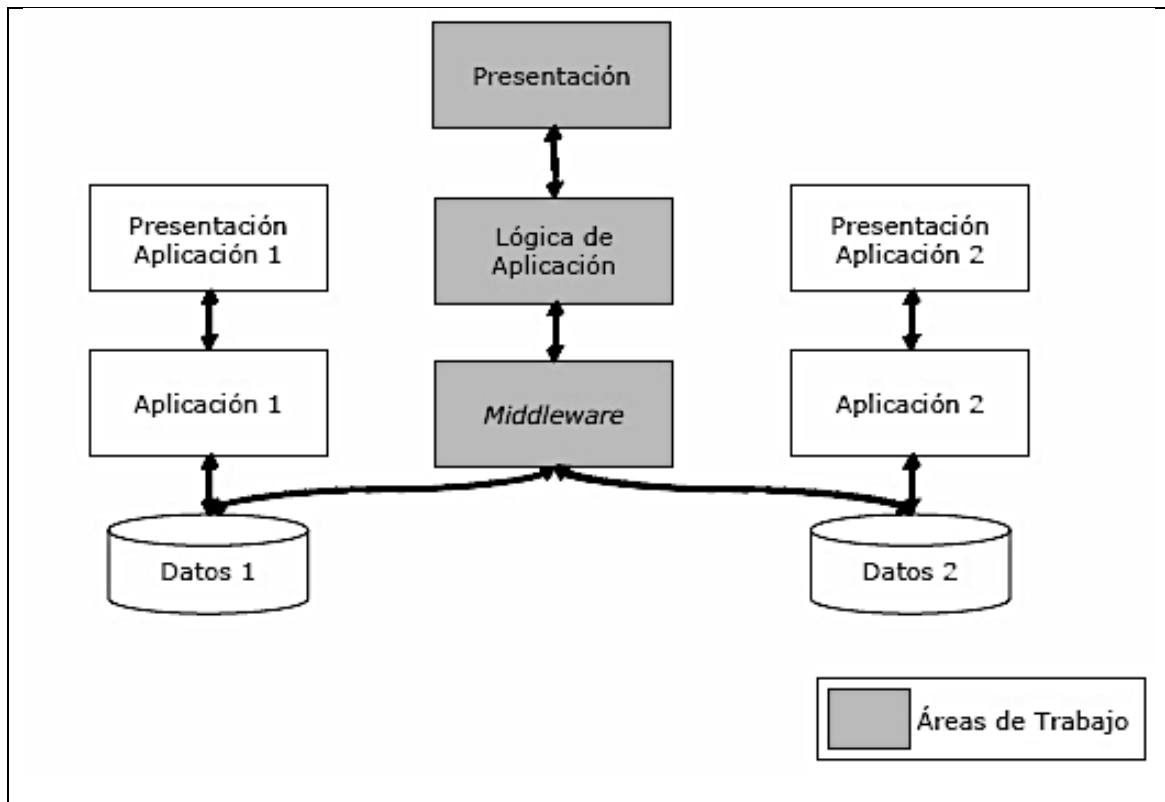


Autor: Melendi David. Universidad de Oviedo.

2.1.2. Integración de la capa de datos.

Este modelo de integración es viable cuando se tiene acceso directo a los datos empleados por las diferentes aplicaciones. Este modelo de integración permite combinar los datos de varias aplicaciones, se puede extraer datos de un sistema para insertarlos en otro sistema independientemente del formato en el que se encuentre. La integración de la capa de datos se la puede realizar de una manera simplificada utilizando herramientas de integración o middleware. Este modelo puede crear una dependencia muy grande del modelo de datos y una necesidad de reestructurar la lógica del negocio. Este modelo es orientado tradicionalmente a sistemas que concentran información y distribuyen actualizaciones.

Figura P : Integración de la capa de datos

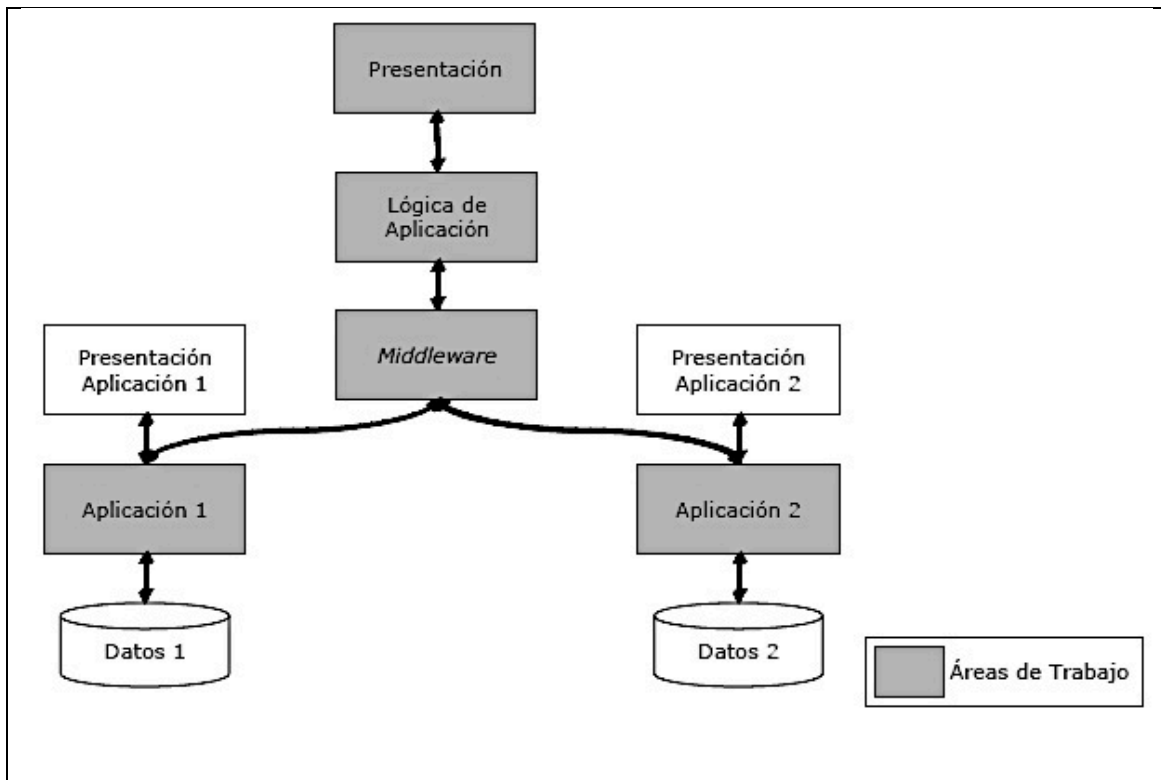


Autor: Melendi David. Universidad de Oviedo.

2.1.3. Integración funcional.

Los modelos de integración de presentación y de datos no permiten conocer, ni manipular lo referente a la lógica del negocio, es por este motivo que se ha desarrollado el modelo de integración funcional, con el cual es posible la integración de la lógica del negocio mediante el uso de herramientas Middleware orientado a mensajería, servicios web y monitores transaccionales.

Figura Q : Integración funcional



Autor: Melendi David. Universidad de Oviedo.

Existen diferentes aproximaciones para realizar este tipo de integración, que dependerán del manejo de la aplicación por parte de la empresa, de la arquitectura utilizada por la empresa, también dependerá el enfoque de la aplicación o de la utilización o no de procesos de negocio. Es por lo antes mencionado que una integración funcional se la realiza a través del código de la aplicación original, siempre y cuando se disponga de los API's⁴¹ de la aplicación.

Para realizar una integración funcional también se disponen herramientas de interface entre aplicaciones de tipo de invocación de procedimientos remotos, sistemas de mensajería y monitores transaccionales.

Invocación de procedimientos remotos.

- RPC. Se construyen procesos que pueden ser ejecutados con la ayuda o a través de sockets⁴².

⁴¹ Application Programming Interface, código fuente utilizado como interface por los componentes de software.

⁴² Punto de comunicación entre procesos.

- Sistemas de objetos distribuidos. Se puede acceder a objetos de forma remota.

Sistemas de mensajería.

- Middleware orientado a mensajes. Se basa en una comunicación asíncrona que permite el pase de mensajes entre aplicaciones.
- Web Services. Basado en estándares abiertos como http, XML, SOAP.

Monitores transaccionales.

Permiten la gestión de transacciones entre aplicaciones.

2.1.3.1. *Aproximaciones de la integración funcional.*

Existen tres aproximaciones para realizar la integración funcional de acuerdo al escenario de integración: consistencia de datos, procesamiento multipaso y componentes plug & play (conectar y usar).

Consistencia de datos. El objetivo de la aproximación por consistencia de datos se basa en la coordinación de las fuentes de datos mediante aplicaciones integradas. Este escenario es ideal cuando existen varias aplicaciones ya desarrolladas existiendo datos duplicados en dichas aplicaciones en la cual la actualización de un dato afecta a varias fuentes de información.

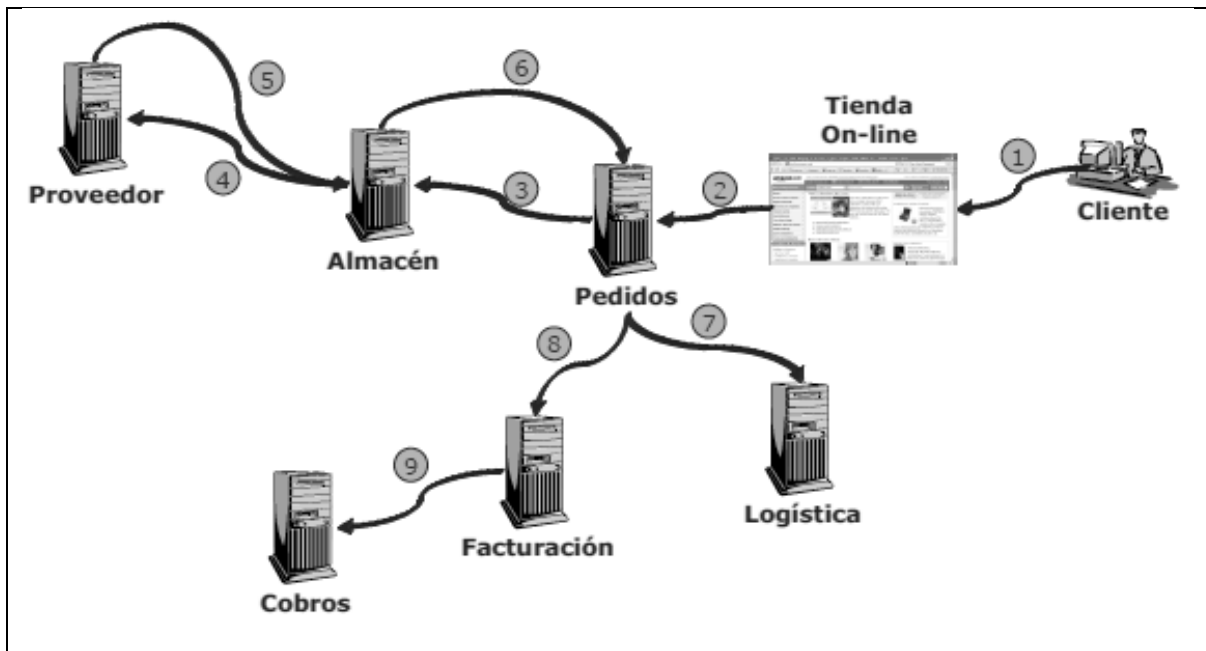
Figura R : Integración Funcional por consistencia de datos



Autor: Melendi David. Universidad de Oviedo.

Procesamiento multipaso. Este tipo de integración se enfoca a la coordinación entre aplicaciones ya que en un proceso de negocio intervienen varios pasos para completar una tarea, es por ello que el procesamiento multipaso mantiene el orden lógico de un proceso o de varios procesos, gestionando los flujos de trabajo y evitando que haya intervención humana en este orden sino que sea un proceso totalmente automatizado y así también evitando la reescritura de información de manera manual.

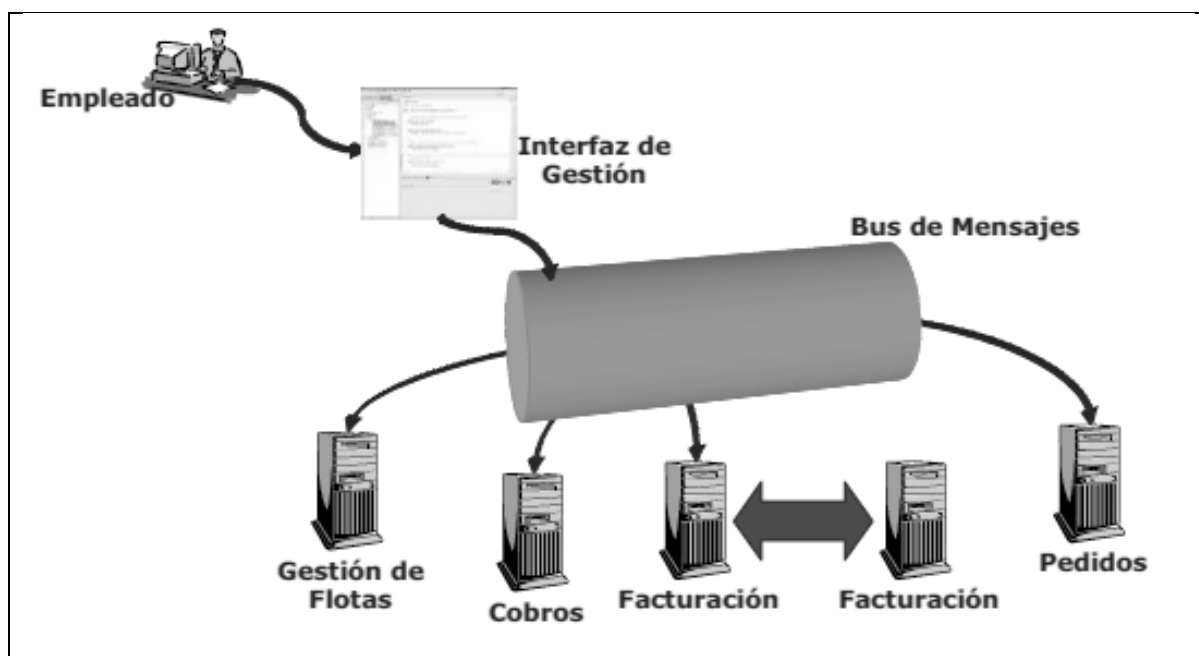
Figura S : Integración funcional por procesamiento multipaso



Autor: Melendi David. Universidad de Oviedo.

Componentes plug & play (conectar y usar). En este tipo de integración se realizan interfaces definidas para el intercambio de información entre aplicaciones, tomando en cuenta la flexibilidad de la arquitectura, es decir; que se pueda realizar cambios sin afectar el funcionamiento del resto de aplicaciones. En esta integración se manejan componentes altamente sustituibles y conectores capaces de administrar las conexiones entrantes como salientes.

Figura T : Integración funcional por componentes plug & play



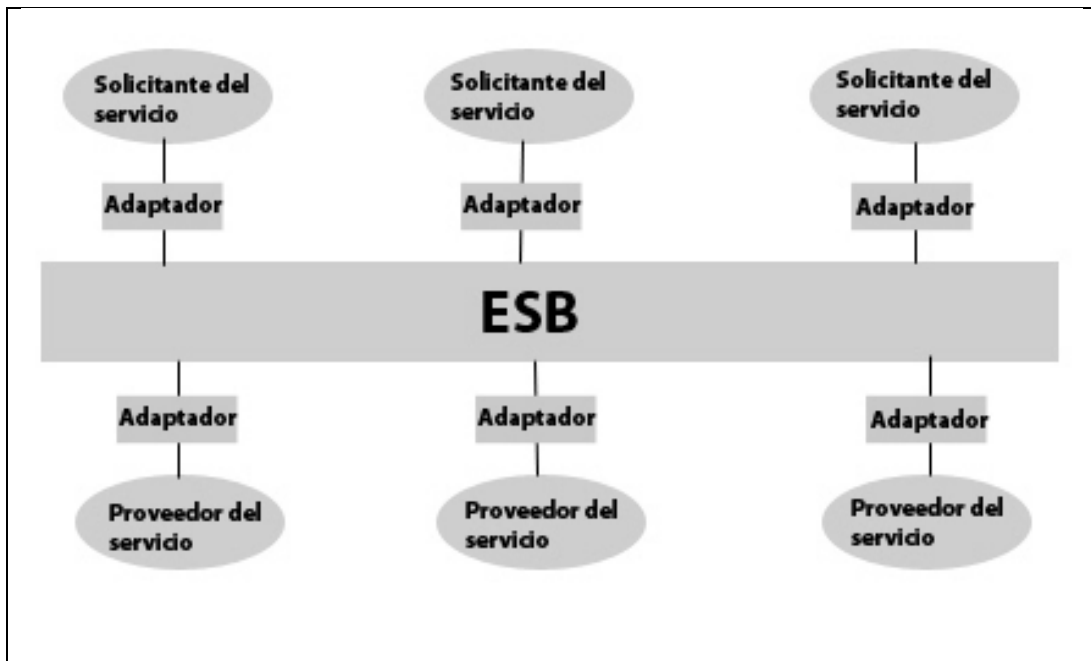
Autor: Melendi David. Universidad de Oviedo.

2.2. Funcionamiento de las herramientas para integración de sistemas con la arquitectura SOA.

Las herramientas de integración estudiadas requieren de un número de requerimientos que se deben implementar en la infraestructura del Bus de integración. Para que una integración de sistemas tenga éxito se la debe realizar en tiempo real, y que cualquier cambio que se deba realizar en una de las aplicaciones no interfiera en el correcto funcionamiento de la arquitectura de integración.

La infraestructura que se debe llegar a lograr en una integración de sistemas se la representa en la figura que se encuentra a continuación, en donde se tiene un conjunto de servicios unidos por el ESB (Bus de servicios empresariales) y a los cuales se puede acceder por medio de un adaptador.

Figura U : Estructura de integración



Autor: Juan Pablo Almeida Durán

Esta arquitectura de integración debe proveer una transferencia segura de transacciones entre los diferentes actores del proceso. Debe tener la capacidad de conversión de datos que son comunicados vía mensajería XML, las transacciones deben ser ruteadas dinámicamente y manejar un registro de información estadística de las transacciones.

La infraestructura de integración necesita la gestión de cada aplicación de excepciones y, además; extenderla para soportar funciones propias del Bus de Servicios Empresariales, para ello la herramienta integradora debe poseer un monitoreo de las acciones basado en web para su fácil acceso, registro de eventos del ESB, al igual que el registro de eventos por parte de los servicios a los que se está integrando, se debe registrar también los errores en la capa de los flujos del negocio.

Al enfrentarnos a una integración de aplicaciones basadas en la arquitectura SOA es mandatorio la modularidad y desacoplamiento de las funciones de negocio, como fue mencionado en el capítulo anterior. Es mandatorio también que los servicios se puedan mezclar y ser combinados para crear nuevos servicios compuestos, todo esto con las herramientas de integración.

La integración de aplicaciones debe ocultar la programación interna del servicio así como los detalles de la implementación y solo exponer la funcionalidad expuesta para ser consumida por otra aplicación, a este requerimiento se lo conoce como encapsulamiento, ya que se separan los datos y la lógica del negocio de la interfaz que se desea exponer, para esto se debe minimizar las dependencias, ya que el bajo acoplamiento es la garantía de que los servicios de SOA no se vean afectados al realizar cambios de las implementaciones en los proveedores de los servicios.

La seguridad de las transacciones en el bus de servicio empresarial debe contar con un acceso seguro a la conexión de la red de servicios, que la información que se transporta entre capas sea íntegra con medidas diseñadas de tal forma que prevengan transmisiones fraudulentas, validando las transmisiones de datos. Se debe asegurar que se puedan rastrear las transacciones al nivel de entidad.

Las herramientas de integración deben poseer un componente de monitoreo de la disponibilidad de la infraestructura que soporta la solución, esto para mantener alta la disponibilidad y capacidad de reacción ante eventualidades así como mantener la trazabilidad de los servicios solicitados, es por ello que las herramientas de integración poseen una base de datos para el manejo estadístico de las transacciones. Al escoger una herramienta ESB se debe asegurar que sea altamente parametrizable para reducir la aparición de errores de programación y así simplificar las pruebas y además para simplificar la administración de la infraestructura.

ESB a través de un lenguaje estándar como BPEL⁴³ debe habilitar la opción de coordinar y organizar la invocación de los diferentes servicios SOA, para así interactuar con la infraestructura de los procesos del negocio de la arquitectura. Un ESB debe traducir el dato transportado para así tener un mapeo de datos exitoso.

Un ESB debe tomar en cuenta que muchos de los procesos expuestos para ser consumidos pueden tener una duración de días o semanas por lo cual debe

⁴³ Business Process Execution Language, por sus siglas en inglés, es un lenguaje ejecutable para especificar acciones entre procesos de negocio y servicios web.

ofrecer la funcionalidad de ejecutar las tareas de una manera paralela y asincrónica.

2.3. Como realizar integración de sistemas con las herramientas de integración disponibles.

Cuando se enfrenta a la implementación de una integración de aplicaciones se debe tener en cuenta la manera en que se realiza según la arquitectura SOA, en la cual el sistema de mensajería es el primordial.

2.3.1. Mensajería.

Cuando dos sistemas requieren comunicarse, esa solicitud y su respuesta se lo hará a través de mensajería. Una petición realizada a través de mensajería posee:

Cabecera. En la cabecera del mensaje se listan los campos que sirven tanto al emisor como al receptor para identificar el mensaje. Los campos que posee la cabecera son:

- **ID.** Este campo permite localizar e identificar al mensaje y es un procedimiento que se realiza el momento del envío.
- **Destino.** La dirección en la cual se encuentra el elemento receptor del mensaje.
- **Expiración.** Este parámetro indica el tiempo de validez que el mensaje tendrá.
- **Prioridad.** Dado por un valor en el rango de 0 a 9 corresponde la prioridad del mensaje donde 9 es el mensaje con mayor prioridad.
- **Respuesta.** Este campo especifica la dirección a la cual se enviará la respuesta del mensaje.

Cuerpo. En este elemento se especifica como las partes del mensaje aparecen, pueden ser definiciones abstractas, tipos serializados, o formas de codificación. En el cuerpo de un mensaje aparece la manera en que se ensamblarán los mensajes.

Dependiendo de la herramienta Middleware que se utilicen, las colas de mensajes serán comunes en la arquitectura y de acuerdo a ello se tienen los tipos de mensajería.

- Petición/respuesta. El cliente realiza una petición de un servicio a un proveedor, normalmente estas peticiones se las hace por el protocolo más usado, http.
- Solo petición. En este tipo de mensajería no es necesaria una respuesta inmediata por parte del proveedor del servicio. Quien solicita el servicio enviará la petición y esta se almacenará en una cola de mensajes para una posterior respuesta del proveedor del servicio.
- Publicación/Suscripción. Permite a una aplicación publicar sus mensajes en el servicio, para así recibir las peticiones de todas las aplicaciones que se encuentren suscritas al servicio.

Para lograr una estructura de mensajería independiente de las plataformas usadas y que se acoplen a una arquitectura SOA y sus requerimientos, se debe utilizar un Middleware orientado a mensajería.

Lo que un Middleware va a lograr en una infraestructura de integración SOA es la definición de clientes y servidores para el intercambio de mensajes entre plataformas, permitir el proceso de los mensajes con la posibilidad de administración de parte del usuario, que a pesar de que una de las aplicaciones no se encuentre disponible el mensaje sea distribuido, evitar la pérdida y duplicación de datos transmitidos.

2.3.2. Conectores.

Cuando en una empresa se presenta el caso que una aplicación desarrollada de manera homogénea, y que no entra dentro de los requerimientos de una SOA se debe recurrir a una solución que desacople a esta aplicación y la haga accesible dentro de la arquitectura, es ahí cuando se utilizan los conectores o adaptadores. Estos adaptadores son invocados mediante peticiones SOAP y se encargan de

interactuar con el sistema “cerrado” para así hacer efectiva la ejecución de sus servicios⁴⁴.

Los adaptadores o conectores son piezas de software de poca complejidad que interactúan entre el sistema y el bus de servicio empresarial ESB ingresando datos y obteniendo resultados de esta invocación, estos adaptadores logran un gran nivel de integración y hacen que un sistema “cerrado” se comporte como un componente más de la arquitectura SOA.

Existen 2 tipos de conectores, los que sirven para que la aplicación cliente haga la petición al ESB y los que sirven para que el ESB obtenga un resultado de una aplicación “cerrada”.

2.4. Herramientas disponibles para la realización de interfaces entre aplicaciones utilizando la arquitectura orientada a servicios.

Las herramientas de software que se encuentran disponibles para realizar interfaces entre aplicaciones se someten a los tipos de licencia existente, como lo son el software propietario, el software libre y de código abierto. En este capítulo se enumerará una serie de estas herramientas que servirán para escoger el software a ser analizado de acuerdo al planteamiento en el plan de tesis (3 herramientas de integración en la que se incluya 1 con licencia libre).

2.4.1. Herramientas para la realización de interfaces entre aplicaciones utilizando la arquitectura orientada a servicios, bajo el modelo de licenciamiento propietario.

A continuación se lista una serie de herramientas de integración de aplicaciones que entran dentro de los parámetros de la presente investigación:

- Compatibilidad con los 3 motores de bases de datos más vendidos.
- Compatibilidad con la arquitectura orientada a servicios SOA.
- La forma de licenciamiento es propietario.


⁴⁴ Puntos de entrada de SOA. IBM. (2010). <http://www.ibm.com/developerworks/ssa/webservices/newto/index.html>

2.4.1.1. Microsoft® Biztalk® Server

| Microsoft® Biztalk® Server | |
|--|---|
| http://www.microsoft.com/biztalk/en/us/default.aspx |  |
| <p><i>“BizTalk Server es la solución de servidor de Microsoft para la integración e interconexión de sistemas. BizTalk Server 2010 es un producto maduro, ya por su séptima versión, que permite interconectar sistemas heterogéneos. Incluye más de 25 adaptadores multi-plataforma y una potente infraestructura de mensajería con la que consigue conectar entre sí sistemas corporativos tanto dentro del ámbito de la empresa como entre éstos y sistemas externos (clientes, proveedores, administraciones públicas, etc). Además de sus funciones de integración, BizTalk proporciona un servicio de mensajería perdurable y potente, un motor de reglas de negocio, conectividad EDI, Monitorización de Actividad de Negocio (BAM, Business Activity Monitoring), gestión de RFID y conexión con entornos de host y mainframe IBM”⁴⁵.</i></p> | |

Tabla 3 : Microsoft® Biztalk® Server

2.4.1.2. Oracle® Fusion Middleware®

| Oracle® Fusion Middleware® | |
|--|---|
| http://www.oracle.com/us/products/middleware/index.html |  |
| <p>“Oracle está invirtiendo en Oracle Applications, Oracle Fusion Middleware 11g para facilitar la integración de aplicaciones. Oracle Application Integration Architecture (AIA) está creada a partir de Oracle Fusion Middleware 11g y está diseñada para acelerar los proyectos de integración.</p> <p>Oracle cuenta con un completo conjunto de herramientas abiertas y punteras en el sector que ofrecen integraciones sostenibles en aplicaciones Oracle y de otros proveedores. Oracle's Fusion Middleware 11g proporciona un completo conjunto</p> | |

⁴⁵ Introducción a Microsoft® Biztalk® Server. Microsoft Corporation®. <http://www.microsoft.com/es-es/biztalk/overview.aspx>

de herramientas para diferentes estilos de integración:

Integración de datos: Oracle proporciona integración de datos en tiempo real de clase empresarial tanto en aplicaciones Oracle como de otros proveedores.

Integración de proceso: facilita la creación, implantación y gestión de arquitectura orientada al servicio (SOA) con la mejor tecnología completa, abierta e integrada de su categoría. Oracle proporciona productos pioneros en el sector para SOA a través de Oracle SOA Suite y para la gestión de procesos empresariales (BPM)

Portal, interacción de usuarios y Enterprise 2.0: Oracle ofrece la plataforma de portales empresariales más lograda, abierta y fácil de usar del sector que hace posible que tanto los desarrolladores como los usuarios profesionales puedan crear, instalar y desarrollar aplicaciones y portales compuestos de próxima generación”⁴⁶.

Tabla 4 : Oracle Fusion Middleware

2.4.1.3. IBM® WebSphere Software®

| IBM® WebSphere Software® | |
|--|--|
| http://www-01.ibm.com/software/websphere/ |  |
| <p>“WebSphere es la plataforma de integración de software de IBM. Incluye toda la infraestructura de middleware, como servidores, servicios y herramientas; necesaria para grabar, ejecutar y supervisar 24 horas al día, 7 días a la semana, aplicaciones Web de nivel industrial y soluciones para varias plataformas y varios productos. WebSphere proporciona un software de integración confiable, flexible y robusta.</p> <p>WebSphere Application Server es la base de la infraestructura; todo lo demás funciona basado en él. Soporta entornos que son SOA y que no lo son. WebSphere Process Server, que se basa en WebSphere Application Server, y WebSphere Enterprise Service Bus, establecen la base para aplicaciones</p> | |

⁴⁶ Introducción a Oracle® Fusion Middleware®. Oracle Corporation. <http://www.oracle.com/es/products/applications/application-integration-architecture/index.html>

modulares con arquitectura orientada a servicios (SOA). Colectivamente, soportan el uso de normas empresariales para impulsar aplicaciones que soportan los procesos empresariales. Los entornos de alto rendimiento también usan WebSphere Extended Deployment como parte de su infraestructura básica. Otros productos WebSphere proporcionan una amplia variedad de servicios adicionales, según se describe a continuación.

CICS Transaction Server actualmente es la base de la mayoría de las aplicaciones de sistema principal y soporta el desarrollo de aplicaciones en lenguajes muy utilizados, como COBOL, PL/I, C/C++ y Java.

WebSphere es una plataforma modular basada en estándares abiertos soportados por la industria. Puede conectar sus activos ya existentes a WebSphere a través de interfaces confiables y perdurables; también puede seguir ampliando su entorno a medida que las necesidades aumentan. WebSphere funciona en varias plataformas, como AIX, HP Unix, i5/OS, Linux, Sun Solaris, Windows y z/OS⁴⁷.

Tabla 5 : IBM® WebSphere Software®

2.4.1.4. SAP NetWeaver®

| SAP NetWeaver® | |
|---|---|
| http://www.sap.com/spain/platform/netweaver/index.epx |  |
| <p>“SAP NetWeaver constituye la base técnica de las aplicaciones SAP. Además, incluye una cartera de tecnología empresarial que le permite ampliar sus aplicaciones para llegar a más personas y adoptar nuevos procesos, dispositivos y modelos de consumo. SAP NetWeaver le ayuda a mejorar la productividad de su equipo y la integración de la empresa. También le permite simplificar y gestionar su entorno de TI, y reduce los costes operativos para que los recursos de TI puedan centrarse de lleno en la innovación.</p> <p>Con SAP NetWeaver, su organización de TI puede crear aplicaciones dispares y hacer que los sistemas funcionen conjuntamente para integrar de forma completa los procesos</p> | |

⁴⁷ Introducción a IBM® WebSphere Software®. IBM Corporation. <http://www.ibm.com/developerworks/ssa/websphere/newto/index.html>

empresariales. Esta solución intercambia información, ejecuta las transacciones de manera fluida y hace que trabajen como si estuvieran en un único sistema”⁴⁸.

Tabla 6 : SAP NetWeaver®

2.4.1.5. **Informatica® PowerCenter®**


| Informatica® PowerCenter® | |
|--|---|
| http://www.informatica.com/us/products/enterprise-data-integration/powercenter/ |  |
| <p>“Informatica® PowerCenter® es una plataforma global y unificada de integración de datos que permite acceder, descubrir e integrar datos de prácticamente cualquier sistema y en cualquier formato, para distribuirlos por toda la organización a la velocidad deseada.</p> <p>Gracias a su alta disponibilidad, elevado rendimiento y completa escalabilidad, PowerCenter sirve de base para todos los proyectos de integración de datos e iniciativas de integración empresarial que se llevan a cabo en todo el negocio. Por ejemplo:</p> <ul style="list-style-type: none"> • Migración y consolidación de datos. • Sincronización y replicación de datos. <p>Data warehouses empresariales y operational data stores.</p> <ul style="list-style-type: none"> • Integración de datos en toda la empresa. • Gobierno de datos. • Arquitecturas Orientadas a Servicio (SOA) • Centros de competencia de integración. • Gestión de datos maestros”⁴⁹. | |

Tabla 7 : Informatica® PowerCenter®

⁴⁸ Introducción a SAP NetWeaver®. SAP AG.

<http://www.sap.com/spain/platform/netweaver/index.epx>

⁴⁹ Informatica® PowerCenter. Informatica

http://www.informatica.com/INFA_Resources/br_powercenter_6659es.pdf

2.4.1.6. **WebMethods®**



| WebMethods® | |
|---|---|
| http://www.softwareag.com/corporate/products/wm/default.asp |  |
| <p>“WebMethods posee ventajas basadas en SOA, en la integración flexible y en el nivel de procesamiento, allanando el camino para satisfacer las necesidades técnicas de los negocios de TI en el futuro.</p> <p>Sólo una integración independiente y una plataforma de procesos permitirán un diseño rápido en toda la compañía y la ejecución en tiempo real de las soluciones de control de los procesos de negocio. Con las capacidades mejoradas de webMethods 8.2 Software AG está ofreciendo una gran cantidad de ventajas que ayudarán a nuestros clientes a ampliar y acelerar sus implementaciones de webMethods.</p> <p>Las organizaciones están demostrando un mayor interés en tener una mejor alineación de sus roles, a través del uso integrado de sus suites que apoyan el modelo para los procesos de negocios, y orientado a objetivos y análisis de las bases de datos y diseño, así como los requisitos de gestión. Por ejemplo, las organizaciones están definiendo, como una buena práctica, los procesos de negocios, eventos y flujos de trabajo en las herramientas BPA, y trasladando esta información a las herramientas del modelado TI y BPMS, produciendo software, servicio de análisis de datos, diseño y desarrollo más especializados”⁵⁰.</p> | |

Tabla 8 : WebMethods®

2.4.1.7. **TIBCO® Business Integration®**

| TIBCO ® Business Integration | |
|--|---|
| http://www.tibco.com/products/soa/business-integration/default.jsp |  |
| <p>“Tibco Business Works es una EAI (Enterprise Application Integration) y un BSE (Bus de Servicios de Empresa). Facilita la integración de aplicaciones</p> | |

⁵⁰ Introducción a WebMethods®. Software AG.
http://www.softwareag.com/corporate/products/wm/application_integration/default.asp

empresariales en ambientes técnicamente heterogéneos (Mainframe IBM, conjunto de servicios web Java, ...).

Este producto nos permite “diseñar” procesos técnicos que contienen “actividades” diversas: acceso a datos, envío de mensajes JMS en un bus de mensajes (MOM), petición de servicios (web) en un bus de empresa (BSE), etc... Este procedimiento visual nos permite acelerar la implementación de las especificaciones funcionales y técnicas gracias a una mejor visibilidad de ese “código” programado al diseñar.

Es un software indispensable que permite crear un bus de mensajes JMS especialmente pensados para el mundo TIBCO / J2EE. EMS también propone otros API de conexión accesibles con otros lenguajes (C/C++/C#/...).

Este software de BPM (gestión de procesos de negocio, por sus siglas en inglés), sucesor de Staffware, nos permite integrar los procesos de la empresa en el corazón del BSE. Esto nos permite definir las etapas de un proceso de empresa de la misma manera que los equipos de estrategia conciben todo sobre papel y pueden añadir o cambiar etapas sin invalidar el proceso existente. Este producto nos permite concentrarnos en el proceso y no en la técnica. Cada etapa de iProcess se puede alternar con los servicios que ofrece, por ejemplo, por TIBCO BW⁵¹.

Tabla 9 : TIBCO® Business Integration®

2.4.1.8. Sonic ESB®

| | |
|---|--|
| Sonic ESB® | |
| http://www.progress.com/en/sonic/sonic-esb.html |  |
| <p>“Sonic ESB® es un bus de servicios empresariales que facilita la integración y la reutilización flexible de componentes empresariales mediante una arquitectura basada en estándares y orientada a servicios (SOA).</p> <p><u>Conexión</u></p> | |

⁵¹ Introducción a TIBCO® Business Integration®. TIBCO. <http://www.tetra-si.com/es/tibco-eai-bse-bpm-4/>

Conecta aplicaciones, servicios web y cientos de tipos de diferentes sistemas heredados de manera sencilla. Proporciona comunicaciones robustas, fácilmente escalables y seguras. Enlaza servicios y procesos por toda la compañía sin fisuras.

Mediación

Concilia protocolos incompatibles, formatos de datos y modelos de interacción de servicios conectados al ESB. Elimina dependencias entre servicios, inflexibles y codificadas dentro de estos servicios. Combina y amplía servicios existentes para alcanzar nuevos requisitos.

Control

Despliega y actualiza de forma dinámica los servicios almacenados en el ESB. Configura procesos de negocio y mediaciones entre los servicios. Gana visibilidad para toda la infraestructura distribuida”⁵².


Tabla 10 : Sonic ESB®

2.4.2. Herramientas para la realización de interfaces entre aplicaciones utilizando la arquitectura orientada a servicios, bajo el modelo de licenciamiento libre.

A continuación se listará una serie de herramientas de integración de aplicaciones que entran dentro de los parámetros de la presente investigación:

- Compatibilidad con los 3 motores de bases de datos más vendidos.
- Compatibilidad con la arquitectura orientada a servicios SOA.
- La forma de licenciamiento es libre.

2.4.2.1. Mule ESB Community Edition

| Mule ESB Community Edition | |
|---|--|
| http://www.mulesoft.org/download-mule-esb-community-edition |  |
| Mule es un framework de mensajero ESB (Enterprise Service Bus). Es un gestor | |

⁵² Introducción a Sonic ESB®. Progress Software.
http://www.sonicsoftware.com/products/docs/sonic_esb_es.pdf

de objetos escalable y distribuible que puede manejar interacciones con servicios y aplicaciones que usan distintas tecnologías de transporte y mensajeo.


Mule fue diseñado para ser liviano y fácilmente embebible en aplicaciones Java y servidores de aplicación o correr como un servidor stand alone. Se integra con un número de frameworks como spring, hivemind y plexus y soporta muchos componentes de transporte y servicio como JMS, SOAP, JBI, BPEL, EJB, AS/400, HTTP, JDBC, TCP, UDP, SMTP, FILE, FTP y más.

Provee poderosas capacidades de ruteo y auditoración de mensajes que son definidos en los Patrones de Integración Empresariales.

El Proyecto Mule también tiene un Contenedor Java Business Integration (JBI/JSR-208) llamado MuleJBI y se distribuye con un IDE basado en Eclipse llamado MuleIDE, el cual es un set de Plug-Ins para Eclipse que sirven para desarrollar, desplegar y gestionar los proyectos.

Tabla 11: Mule ESB Community Edition®

2.4.2.2. Sopera

| Sopera | |
|---|---|
| http://www.sopera.de/en/ |  |

Software de integración que combina una diversidad de sistemas para hacerlos trabajar como un todo. Sopera es altamente escalable y maneja de manera eficiente los cuellos de botella. La seguridad de las transacciones en el bus de servicio empresarial de Sopera cuenta con un acceso seguro a la conexión de la red de servicios, que la información que se transporta entre capas sea íntegra con medidas diseñadas de tal forma que prevengan transmisiones fraudulentas, validando las transmisiones de datos. Asegura el rastreo de las transacciones al nivel de entidad. Sopera gestiona las excepciones y, además; soporta funciones propias del Bus de Servicios Empresariales, para ello la herramienta integradora posee un monitoreo de las acciones basado en web para su fácil acceso, registro de eventos del ESB.

Tabla 12: Sopera

2.4.2.3. *Petals*

| Petals | |
|---|--|
| http://petals.ow2.org/download-petals-esb.html |  |
| <p>Petals es un ESB de código abierto que provee el ambiente idóneo para la construcción de una arquitectura SOA e integración de aplicaciones. Petals elimina la necesidad de una integración punto a punto gracias a su sistema de mensajería. La ventajas de utilizar Petals en su organización son:</p> <ul style="list-style-type: none"> • Orquestación e integración. Ensamblaje rápido y fácil de servicios web, validación e interpretación de BPEL. • Flexibilidad. Integración de servicios ‘en caliente’. • Conectividad y mediación. Conectores para los estándares más utilizados: WSDL, SOAP, REST, POP, SMTP, IMAP, EJB, JDBC. • Integración con Eclipse. • Modular. | |

Tabla 13: Petals

2.4.2.4. *OpenEAI*



| OpenEAI | |
|--|---|
| http://www.openeai.org/live/index.php?p=5&sub=1 |  |
| <p>OpenEAI es una aplicación basada en protocolos de mensajería entre aplicaciones en formato XML para peticiones y respuesta, publicar y suscribir modelos de mensajes para cualquier empresa. El protocolo de mensajería se basa en los principios inherentes a la integración de aplicaciones, lo cual hace que no se encuentre atado a una plataforma específica.</p> <p>Un análisis riguroso de plantillas e instrucciones facilita el proceso de integración mediante OpenEAI, ya que define objetos XML y define el protocolo por el cual van a ser transferidos. Además, OpenEAI foundation provee a los usuarios de los API's de la aplicación. La tecnología nativa de OpenEAI es Java, JMS y XML, pero eso no implica que OpenEAI pueda comunicarse e integrar aplicaciones con diferentes tecnologías.</p> | |

Tabla 14: OpenEAI

2.4.2.5. *Apache ServiceMix*

| Apache ServiceMix | |
|---|--|
| http://servicemix.apache.org/ |  |
| <p>Es un Bus de Servicio Empresarial (ESB) que posee las funcionalidades de una arquitectura orientada a servicios para así crear la infraestructura necesaria para implementar un ESB. Es construido con la especificación Java (JBI) pero con licencia Apache. Sus principales características son:</p> <p>Normalized Message Router (NMR).</p> <p>Elemento central de la arquitectura, también llamado Bus.</p> <p>Se encarga de la recepción, transformación y ruteo de mensajes entre los componentes.</p> | |

JBI Components (Plugins).

Los componentes JBI son aquellos proporcionados por el contenedor. De estos existen dos tipos distintos: binding components (BC) y service engines (SE).

Componentes de transmisión (Binding components).


Se encargan de la transmisión de datos entre el NMR y otros servicios o aplicaciones externas. Existen dos tipos principales: *consumer* y *provider*. Consumer - recibe mensajes de aplicaciones externas y los envía hacia un servicio interno. Provider - envía mensajes desde un servicio interno hacia un servicio externo.

Servicios de procesamiento (Service Engines).

Unidades que realizan tareas de procesamiento de datos. Hay varios tipos: mecanismos de reglas, de transformación XSLT, de Scripts, BPEL, contenedores EJB, entre otros.

Tabla 15: Apache ServiceMix

2.4.2.6. *Apache Camel*

| Apache Camel | |
|---|--|
| http://camel.apache.org/index.html |  |
| <p>El proyecto Apache Camel es un DSL (Domain Specific language) orientado a la implementación de EIP's (Enterprise Integration patterns) que además integra una serie de tecnologías tales como JMS, HTTP, FTP, SOAP, FILE, ...</p> <p><i>Las ventajas de usar Camel son:</i></p> <ul style="list-style-type: none">Reduce el tiempo de creación de flujos de integración.Reduce la complejidad de los flujos de integración y su mantenimiento.Las rutas pueden ser desplegadas en distintos tipos de contenedores java | |

Una ruta en Camel puede entenderse como un recorrido paso a paso de un mensaje. Existe un "listener" que consume el mensaje, un destino que produce la salida y además pueden existir entre medias toda una serie de componentes que procesan y/o redirigen el mensaje. Para la especificación de dichas rutas se pueden utilizar ficheros de configuración (Spring) o el lenguaje Java DSL (Domain Specific Language).

En Camel existe toda una serie de componentes que dan soporte a distintas tecnologías:

- Web Services
- JMS/ActiveMQ
- E-mail
- HTTP
- JBIF
- Java Beans
- Velocity / XSLT
- Jakarta logging

Tabla 16: Apache Camel

2.5. Cuadro comparativo de herramientas para la realización de interfaces entre aplicaciones.

| Herramienta | Licenciamiento | Soporte de bases de datos, aplicaciones o protocolos | Soporte de sistemas operativos |
|-----------------------------------|----------------|--|---|
| Microsoft® Biztalk® Server | Propietario | -Oracle 9.2, 10.1,10.2, 11.1 - MySQL -SQL Server -SAP -IBM DB2 | -Windows 2003 server -Windows Vista -Windows 7 Windows Server 2008 |

| | | | |
|-----------------------------------|-------------|--|---|
| | | -WebSphere | |
| Oracle® Fusion Middleware® | Propietario | -Oracle - MySQL -SQL Server -SAP -IBM DB2 -WebSphere -Sybase | -IBM AIX -Sun Solaris -HP-UX -Windows -Linux |
| IBM® WebSphere Software® | Propietario | -Oracle - MySQL -SQL Server -IBM DB2 -WebSphere -Sybase | -AIX -HP-UX -Linux -Solaris -Windows -z/OS |
| SAP NetWeaver® | Propietario | -Oracle - MySQL -SQL Server -SAP -IBM DB2 -WebSphere | -AIX -HP-UX -Solaris -Windows |
| Informatica® PowerCenter® | Propietario | -Oracle - MySQL -SQL Server -SAP -IBM DB2 | -Windows |

| | | | |
|---|-------------|---|---|
| | | -WebSphere -Sybase | |
| WebMethods® | Propietario | -Oracle -DB2 -SQL Server | -Windows xp -Windows 2003 server -Windows vista -Windows 2008 server -Linux -Solaris -HP-UX -AIX -OSX |
| TIBCO® Business Integration® | Propietario | -Oracle - MySQL -SQL Server -SAP -IBM DB2 -WebSphere | -Windows XP -Windows Vista -Windows 2003 server -Red Hat Linux -SUSE Linux |
| Sonic ESB® | Propietario | -Oracle - MySQL -SQL Server | -Windows -Linux |
| Mule® Community Edition | Libre | -Oracle - MySQL -SQL Server -IBM DB2 | -Windows 7 -Windows Vista -Windows XP -Windows 2000 |

| | | | |
|--------------------------|-------|--|---|
| | | -WebSphere -Sybase -SAP | -Windows 2003 -Linux |
| Sopera | Libre | -Oracle - MySQL -SQL Server | -Linux -Windows xp -Windows 2003 server -Windows 7 |
| Petals | Libre | -Oracle - MySQL -SQL Server -SAP -IBM DB2 -WebSphere -Sybase | -Windows -Linux -Mac OSX |
| OpenEAI | Libre | -Oracle - MySQL -SQL Server | -Windows -Linux |
| Apache ServiceMix | Libre | -Oracle - MySQL -SQL Server -SAP -IBM DB2 -WebSphere -Sybase | -Windows -Linux -Mac OSX -Unix |

| | | | |
|---------------------|-------|--|---|
| Apache Camel | Libre | <ul style="list-style-type: none"> -Oracle - MySQL -SQL Server -SAP -IBM DB2 -WebSphere -Sybase | <ul style="list-style-type: none"> -Windows -Linux -Mac OSX -Unix |
|---------------------|-------|--|---|

Tabla 17 : Cuadro comparativo de herramientas para la realización de interfaces entre aplicaciones

CAPÍTULO III

3. Guía técnica de las herramientas escogidas para interfaces entre aplicaciones y que usan la arquitectura SOA.

Para cumplir con lo estipulado en el plan de disertación de grado, para el análisis se han escogido 3 herramientas, teniendo en cuenta su compatibilidad con bases de datos de mayor uso, la disponibilidad de documentación, y que al menos una de las herramientas sea bajo el licenciamiento de software libre. Estas herramientas son: Microsoft ® Biztalk® Server, Oracle ® Fusion Middleware® y Mule ESB Community Edition.

La guía técnica desarrollada consta de 5 partes siguiendo el orden lógico de implementación del software de integración, desde el requerimiento de integración hasta el mantenimiento de la herramienta una vez que esta ya se encuentre en un ambiente de producción. Es así como la guía está estructurada de

- Introducción de las herramientas escogidas.
- Requerimientos para la instalación.
- Análisis económico.
- Instalación.
- Guía de realización de integraciones con las herramientas de interface.
- Mantenimiento de las herramientas en el tiempo.

3.1. *Introducción histórica de las herramientas escogidas para ser analizadas.*

3.1.1. *Microsoft® Biztalk® Server.*

Microsoft® Biztalk® Server nace de la necesidad de las empresas de utilizar diferentes herramientas de software con diferentes configuraciones para diferentes procesos de una empresa pero que se manejan dentro de un entorno común y que por lo tanto necesitan interactuar entre sí ya que muchos procesos empresariales utilizan datos en común.

En el año 2000 Microsoft lanza la primera versión de Biztalk Server, con la capacidad de editar y reconocer datos XML, posibilidad de ‘mapear’ datos para su

traslado con el formato recibido de entrada para transformarlo en el formato requerido en la salida, también ofrece la posibilidad de rastrear los datos en el tiempo y reconoce los protocolos EDI, http, https, SMTP y FTP.

La siguiente gran actualización se da en el año 2004 con la capacidad de añadir flujos de trabajo y con la integración del entorno .Net, además de presentar Visual Studio 2003 integrado en la solución para servicios de mapeo. En el año 2006 es lanzada una nueva versión en la cual integra Business Activity Services, soporte para el entorno .Net 2.0 y Visual Studio 2005 integrado. En el año 2007 es lanzada otra versión también conocida como 2006 R2 en la cual se realizan cambios en la interface de usuario con la posibilidad de administración de la solución enteramente desde la interfaz de usuario.

En el año 2010 se lanza la última versión de Biztalk Server, con soporte para integración con herramientas basadas en Cloud Computing y redes sociales. Se presenta una versión separada llamada Biztalk Server RFID que posee la capacidad de identificación por radiofrecuencia, con la cual se puede realizar seguimiento del inventario y otros objetos físicos a los largo de un proceso empresarial.

3.1.2. Oracle Fusion Middleware® SOA Suite.

La primera versión de Oracle Fusion Middleware SOA Suite posee la nomenclatura estándar de Oracle, es la versión 10g R1, en la cual se aborda de manera especializada la arquitectura SOA con soporte para el Bus de Servicio Empresarial, tiene la capacidad de lectura e interpretación de datos XML de los mensajes SOA. Con el lanzamiento de la versión R3 del software, se mejora el modo de instalación, haciéndole mucho mas amigable para el usuario, se incluye la capacidad de creación e interpretación de flujos de trabajo, facilidad de manejo de servicios web, compatibilidad con mayor cantidad de protocolos, también se incluye un ambiente de pruebas dentro de la aplicación.

La última versión de Oracle Fusion Middleware SOA Suite es la g11 lanzada en el año 2009, con mejoras de sus versiones anteriores con respecto a las facilidades para los desarrolladores presentando un ambiente de pruebas con opciones renovadas, incluye el ambiente de desarrollo de aplicaciones SOA en JDeveloper,

conectores y adaptadores para diferentes bases de datos y servicios basados en cloud computing.

3.1.3. Mule ESB Community Edition.

Mule ESB Community Edition es presentado el año 2003 con el enfoque de crear una plataforma de integración de aplicaciones que tenga licenciamiento libre y de código abierto, capacidad de correr en múltiples sistemas operativos e integrar diferentes bases de datos, compatibilidad con la arquitectura orientada a servicios SOA. Mule ESB Community Edition presenta también como característica principal un Bus de Servicio Empresarial con capacidad de transporte para múltiples protocolos.

Mule ESB Community Edition maneja datos en formato XML para el transporte por el bus de servicio empresarial con capacidad de interpretación del formato del dato de entrada para transformarlo al formato del dato de salida requerido. Mule presenta 3 versiones a lo largo de su historia, con 7 actualizaciones repartidas para las 3 versiones, durante el desarrollo de esta investigación la última versión es la 3.2, la cual viene con capacidad de desarrollo de flujos de trabajo, integración con cloud computing.

3.2. Análisis de requerimientos que poseen las herramientas escogidas para ser usadas de interfaces entre aplicaciones.

Las herramientas escogidas para analizar en la investigación, al ser herramientas de software poseen requerimientos mínimos y requerimientos óptimos de hardware y además, poseen requerimientos en el entorno de software para su funcionamiento. Se listarán y analizarán estos requerimientos para dar un juicio de valor acerca de las herramientas escogidas.

3.2.1. Microsoft® Biztalk® Server.

La versión analizada de Microsoft® Biztalk® Server es la 2010 del 23 de septiembre, que es la última disponible.

3.2.1.1. **Requerimiento de hardware de Microsoft® Biztalk® Server.**

- Los requerimientos presentados a continuación se dan para un ambiente de producción.
- Un procesador compatible con Intel Pentium

| | Velocidad del reloj | 1 GHz | 900 MHz | 700 MHz |
|---|----------------------------|--------------|----------------|----------------|
| Número de núcleos del procesador | | | | |
| 1 | | X | | |
| 2 | | | X | |
| 4 | | | | X |

Tabla 18 : Requerimientos de Procesador de Microsoft® Biztalk® Server

- Arquitectura de 64 bits.
- 2 GB de memoria RAM.
- 10 GB de disco duro, independientemente del espacio que ocupe el sistema operativo y el resto de requerimientos de software.
- Tarjeta de red
- Monitor VGA con una resolución mínima de 1024 x 768 píxeles.

3.2.1.2. **Requerimiento de software de Microsoft® Biztalk® Server.**

- Microsoft® Biztalk® Server requiere de sistemas operativos Windows, lo cual no implica que las integraciones de aplicaciones requieran el entorno Windows para los sistemas, ya que Biztalk Server nos oferta un Bus de Servicio Empresarial por el cual recorrerán mensajes en formatos independientes de la plataforma.

- Los sistemas operativos Windows en los cuales funciona Biztalk Server son:
 - Windows Server 2008 R2 (Release 2)
 - Windows Server 2008 con Service Pack 2
 - Windows 7
 - Windows Vista con Service Pack 2
- Microsoft information Services (IIS) 7.0 o 7.5
- Entorno Microsoft .NET 4 o 3.5 con Service Pack 3.5
- SQL Server 2008 R2 o SQL Server 2008 SP1
- SQLXML 4.0 con service pack 1

3.2.2. Oracle Fusion Middleware® SOA Suite.

La versión analizada de Oracle Fusion Middleware® SOA Suite es 11gR1 (11.1.1.6.0).

3.2.2.1. Requerimiento de hardware de Oracle Fusion Middleware® SOA Suite.

- La memoria RAM y Swap requerida es de 4GB, divididos preferentemente en 3 GB de memoria RAM y 1 GB de memoria virtual.
- Oracle SOA Suite necesita un espacio en disco dividido de la siguiente manera:

| Producto | Espacio en disco |
|---------------------------|-------------------------|
| Oracle SOA Suite | 4.81 GB |
| Oracle Service Bus | 1.2 GB |
| Oracle Database | 5 GB |
| TOTAL: | 11.01 GB |

Tabla 19 : Requerimiento de espacio en disco duro de Oracle Fusion Middleware SOA Suite

- Los requerimientos de procesador en servidor difieren de acuerdo al siguiente cuadro:

| Plataforma | Servidor | Velocidad del reloj | Procesador |
|---------------------|---------------------|----------------------------|-------------------|
| IBM AIX | RS/6000 | 900 MHz | eServerP5 |
| Sun Solaris | Sun Sparc | 900 MHz | Ultrasparc IV |
| HP-UX (PA Risc) | HP PA RISC (PA 2.0) | 750 MHz | PA RISC PA 8800 |
| HP-UX(IA64/Itanium) | HP Itanium (IA 64) | 1.4 GHz | Intel Itanium 2 |
| Windows | Cualquier | 3 GHz | Intel Xeon MP |
| Linux | Cualquier | 3 GHz | Intel Xeon MP |

Tabla 20 : Requerimiento de Procesador de Oracle Fusion Middleware SOA Suite

3.2.2.2. *Requerimiento de software de Oracle Fusion Middleware® SOA Suite.*

- Java JDK 6.0 actualización 24
- Oracle Fusion Middleware SOA Suite puede ser instalado bajo los siguientes sistemas operativos:
 - SUSE 11
 - IBM AIX 7.1
 - Solaris 11
 - Windows XP Profesional Service Pack 2
 - Windows Vista
 - Windows 7
 - Windows server 2008
 - Windows 2003 Server
- Navegador web:
 - Firefox 3
 - Safari 4 o superior
 - Internet Explorer 7 o superior

3.2.3. Mule ESB® Community Edition.

La versión utilizada de Mule ESB® Community Edition es la v3.2.

3.2.3.1. Requerimiento de hardware de Mule ESB® Community Edition.

El requerimiento de hardware de Mule ESB Community Edition es uno de los puntos más fuertes de este Bus de Servicio Empresarial, presentándose así:

- Un procesador de un solo núcleo de 1.6 GHz.
- 768 MB de memoria RAM.
- 260 MB de espacio en disco duro.
- Tarjeta de red.

3.2.3.2. Requerimiento de software de Mule ESB® Community Edition.

- Mule ESB Community Edition es capaz de correr en distintos sistemas operativos:
 - Windows 7
 - Windows Vista
 - Windows XP Service Pack 2
 - Windows 2000
 - Windows 2003 Server
 - Linux
 - Solaris
 - AIX
 - HP-UX
 - Mac OSX
- Java JDK 1.5 o superior

3.3. Análisis económico de las herramientas de interfaces entre aplicaciones utilizadas.

3.3.1. Análisis económico de Microsoft® Biztalk® Server.

- El licenciamiento de Microsoft Biztalk Server es bajo el modelo de licenciamiento por procesador.
- El número de núcleos del procesador no modifica el precio del licenciamiento.
- Microsoft Biztalk Server utilizará hasta 2 procesadores, pero si se encuentra instalado bajo un sistema operativo Windows que ocupe más de 2 procesadores, el licenciamiento deberá ser del número de procesadores que ocupe Windows ya que la aplicación saca provecho del rendimiento de Windows.
- El costo de licenciamiento por procesador de Microsoft Biztalk Server 2010 es de \$44.228 y \$5.031 adicionales por la extensión RFID que integra a dispositivos móviles

3.3.2. Análisis económico de Oracle Fusion Middleware® SOA Suite.

- La estructura de Oracle Fusion Middleware SOA Suite permite que se puedan reemplazar partes de este paquete con software desarrollado por la competencia de Oracle, por ejemplo se puede reemplazar el servidor de aplicaciones, el bus de servicio empresarial o la base de datos que forma la estructura de la aplicación.
- Por la consideración anterior Oracle utiliza el licenciamiento por número de procesadores y el costo de cada licencia es de \$50.000 cuando todos los componentes usados son de Oracle.
- El costo por procesador es de \$65.000 cuando se utiliza un servidor de aplicaciones de la competencia de Oracle.

3.3.3. Análisis económico de Mule ESB® Community Edition.

- Mule ESB Community Edition es distribuido bajo el licenciamiento CPAL⁵³, por lo que el software no tiene costo.

3.3.4. Cuadro del análisis de costos de las herramientas.

| Herramienta | Tipo de Licenciamiento | Costo por procesador | Detalles adicionales |
|------------------------------------|---|----------------------|--|
| Microsoft Biztalk Server | Propietario | \$44.228 | Se utilizará 2 procesadores. |
| Oracle Fusion Middleware SOA Suite | Propietario | \$50.000 | Si el servidor de aplicaciones NO es de Oracle el costo es de \$65.000 |
| Mule ESB Community Edition | Licenciamiento de software libre CPAL (Common Public Attribution License) | No Aplica | |

Tabla 21 : Análisis de costos de las herramientas

3.4. Análisis detallado del proceso de instalación de las herramientas de interfaces entre aplicaciones.

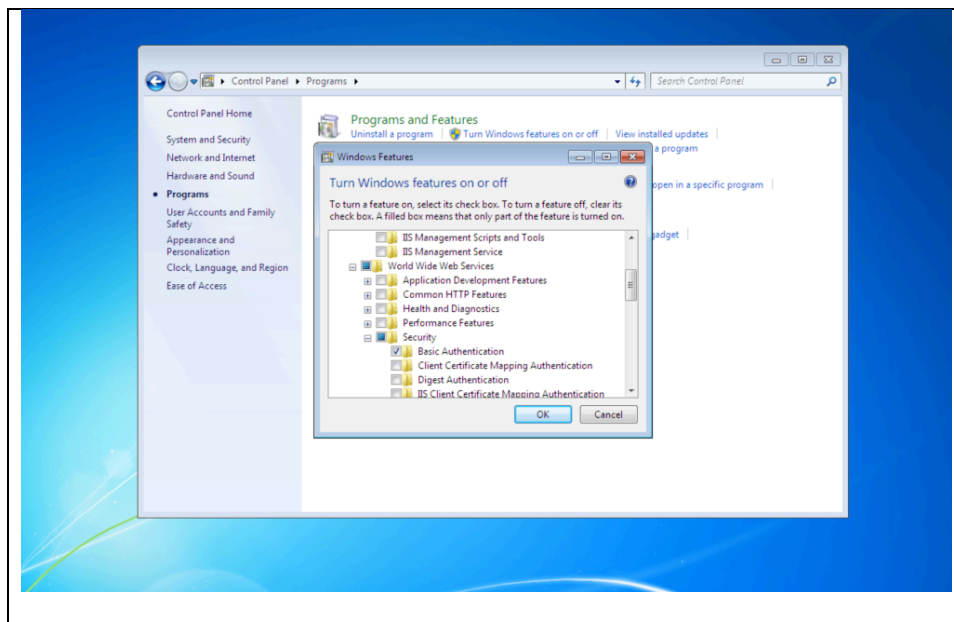
3.4.1. Instalación de Microsoft® Biztalk® Server.

- Para realizar la instalación de Microsoft Biztalk Server se debe preparar el ambiente en el cual se va a ejecutar la aplicación, esto también depende de si el ambiente es de pruebas, desarrollo o producción. En los ambientes de pruebas y desarrollo podemos utilizar las versiones exprés que ofrece Microsoft de herramientas como: SQL Server 2008 o Visual Studio 2010.

⁵³ Common Public Attribution License, Licenciamiento de software libre aprobado por la Open Source Initiative.

- La herramienta de desarrollo Visual Studio 2010 es necesaria tan solo en el ambiente de desarrollo de aplicaciones.
- Es mandatorio que antes de realizar la instalación de Biztalk Server se ejecute la aplicación de actualización del sistema operativo.
- Se debe habilitar en Internet Information Services (IIS) la opción IIS 6 Management Compatibility y en World Wide Web Services la opción Basic Authentication y Windows Authentication.

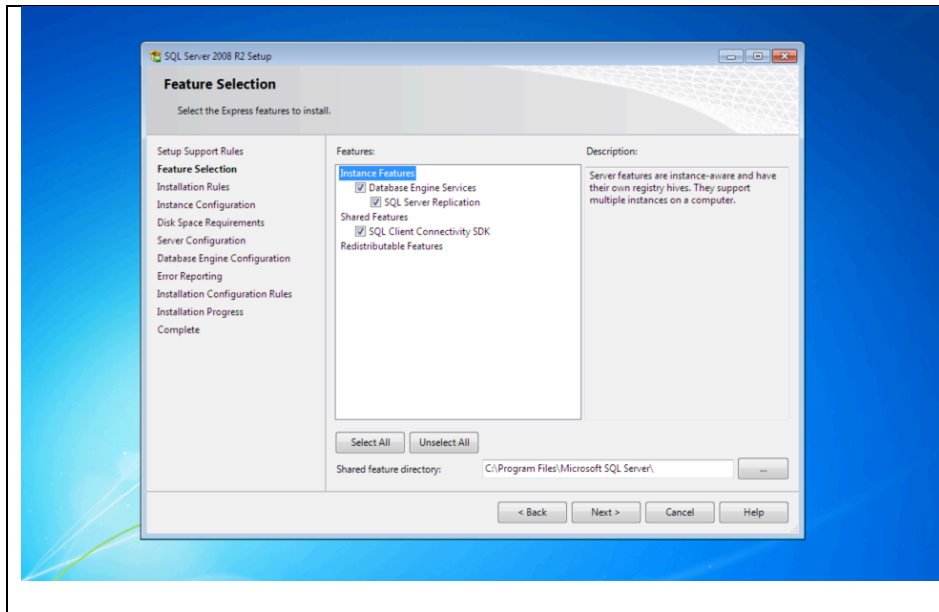
Figura V : Biztalk Server, habilitación de IIS



Fuente: Programa de instalación de Biztalk Server

- Biztalk server requiere de la instalación de SQL Server versión 2008 o posterior.
- La instalación de SQL Server 2008 es estándar y se debe tener en cuenta dos puntos principales:
 - Seleccionar todas las características disponibles de SQL Server 2008.

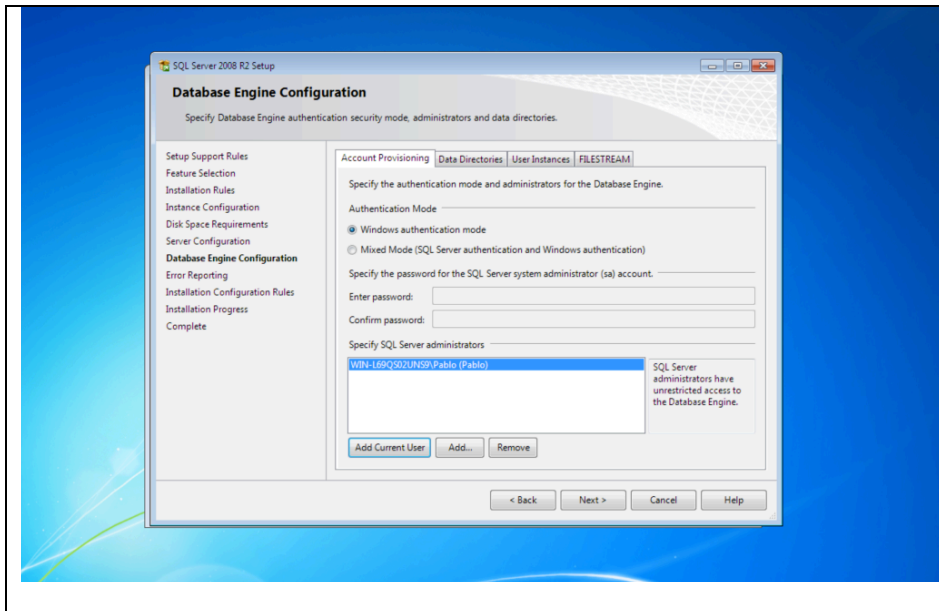
Figura W : Biztalk Server, selección de características de SQL Server



Fuente: Programa de instalación de Biztalk Server

- En la configuración del servidor se escogerá utilizar la misma cuenta para todas las instancias.

Figura X : Biztalk Server, configuración de cuentas de SQL Server

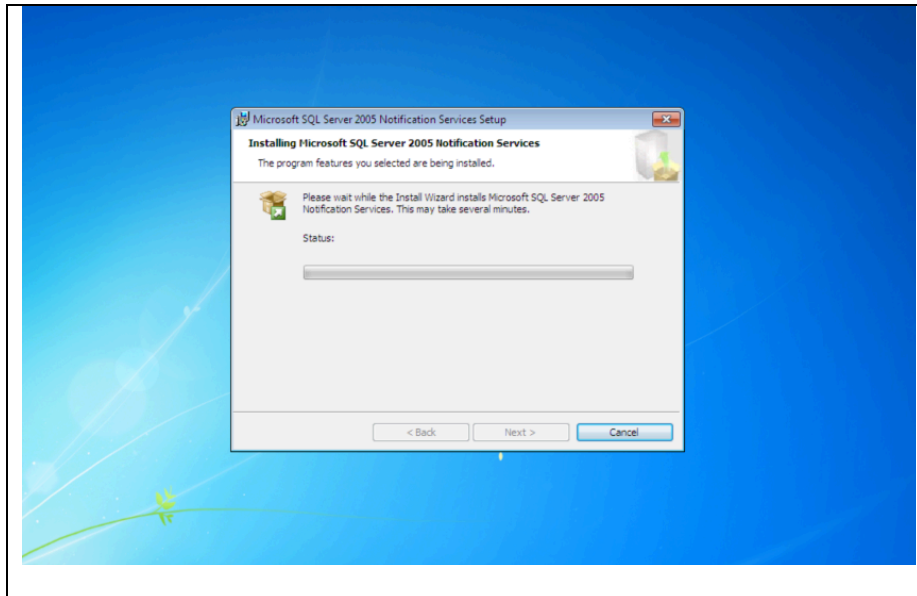


Fuente: Programa de instalación de Biztalk Server

- Una vez instalado SQL Server 2008 se debe añadir un paquete de características adicionales a la instalación las cuales se encuentran en la página <http://www.microsoft.com/en-us/download/details.aspx?id=11988>. Los paquetes que se debe instalar son los siguientes:

- SQL Server Native client (sqlncli.msi)
- SQL Server Objects collection (SQLServer2005_XMO.msi)
- SQL Server Notification Services (SQLServer2005_NS.msi)

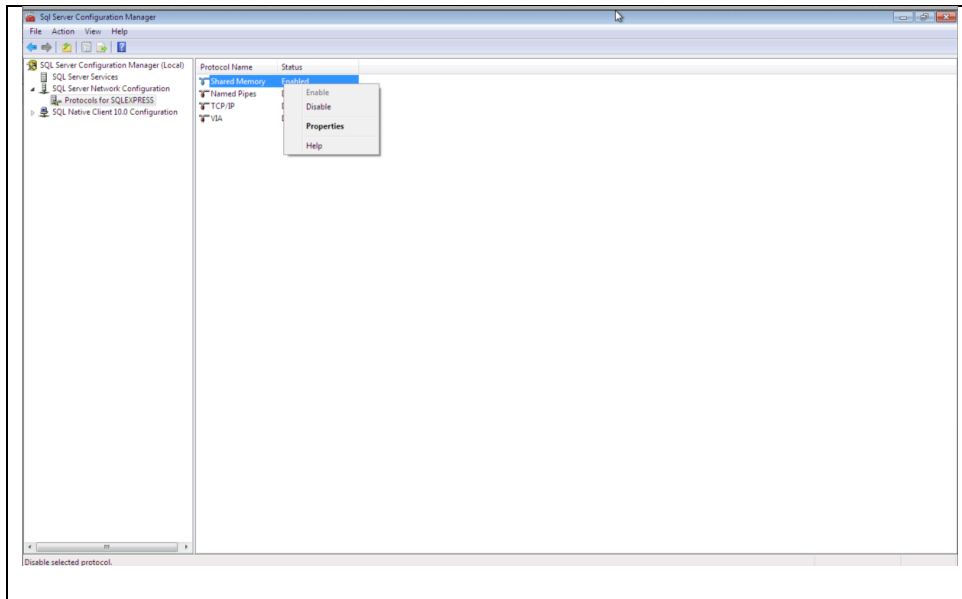
Figura Y : Biztalk Server, instalación de paquetes adicionales s SQL Server



Fuente: Programa de instalación de Biztalk Server

- Se debe deshabilitar el protocolo de memoria compartida en SQL Server 2008. Para ello debemos abrir Configuration Tools , SQL Configuration Manager.
- En SQL Configuration Manager se debe ir a Protocols for MSSQLSERVER, y en la opción Shared Memory se dará click derecho y se escogerá la opción Disable.

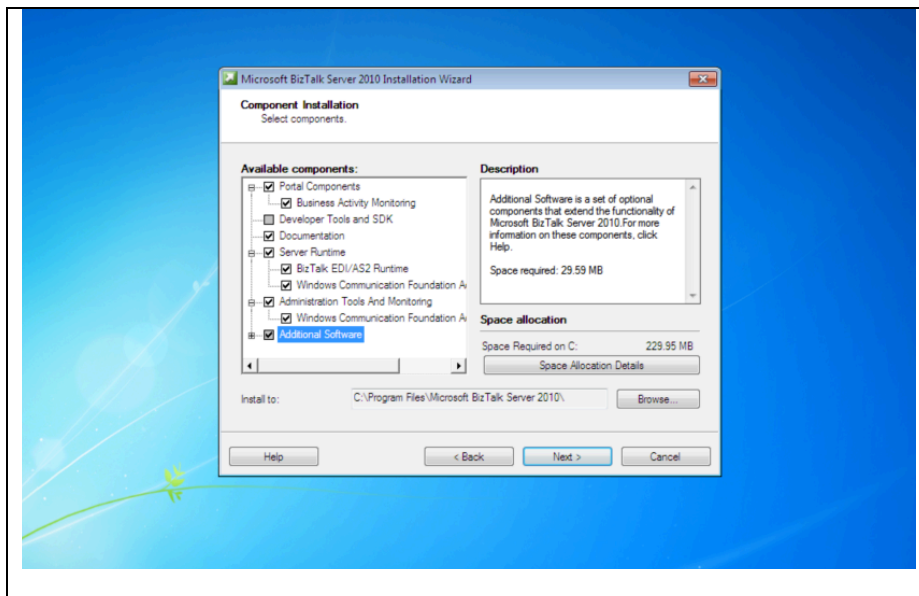
Figura Z : Biztalk Server, deshabilitar la memoria compartida de SQL Server



Fuente: Programa de instalación de Biztalk Server

- Una vez listo el entorno se procede a instalar Microsoft® Biztalk® Server de manera estándar teniendo en cuenta las siguientes consideraciones:
 - En la pantalla de componentes para la instalación, se selecciona todos los componentes disponibles.

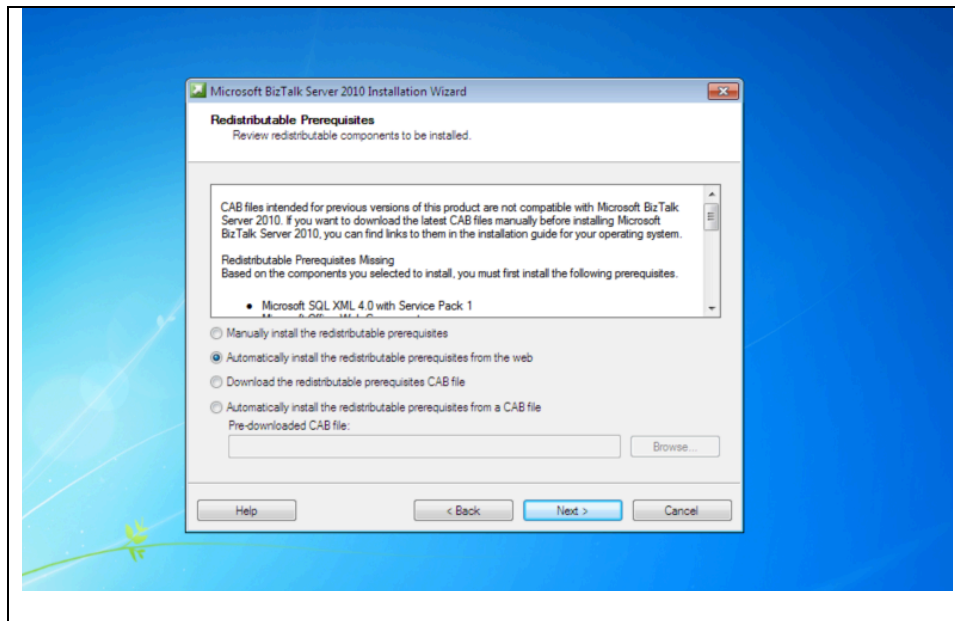
Figura AA : Biztalk Server, selección de componentes



Fuente: Programa de instalación de Biztalk Server

- En la continuación de la instalación se permite que el instalador descargue archivos propios necesarios para la instalación tales como interpretadores de datos XML, interpretadores de mensajes. Esta descarga se realizará al escoger la opción Automatically Install The Redistributable Prerequisites from the web.

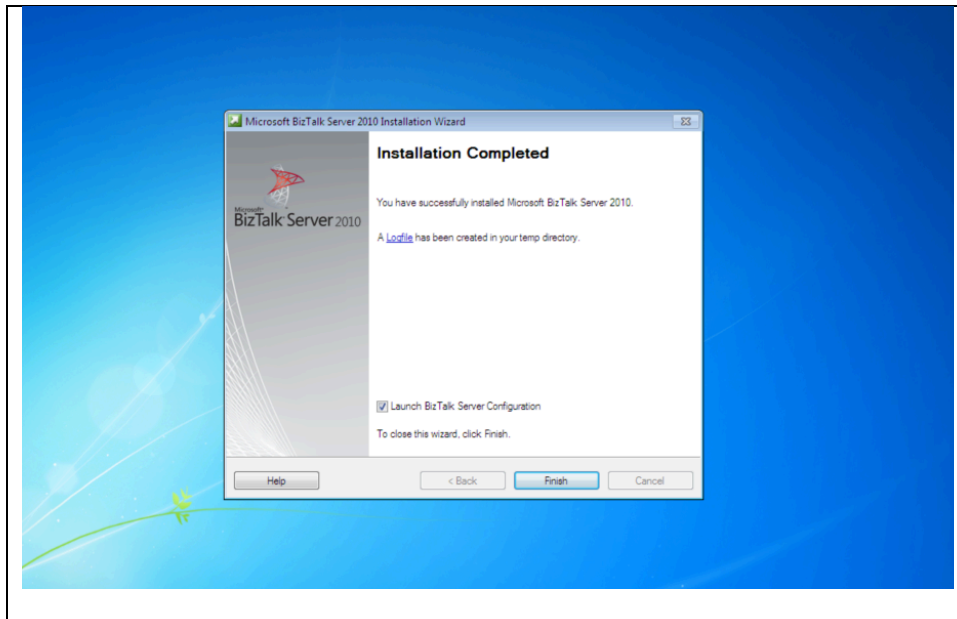
Figura BB : Biztalk Server, descarga automática de características



Fuente: Programa de instalación de Biztalk Server

- Finalizada la actualización de componentes se debe reiniciar el computador y reiniciar el instalador de Biztalk Server, se terminará la instalación.

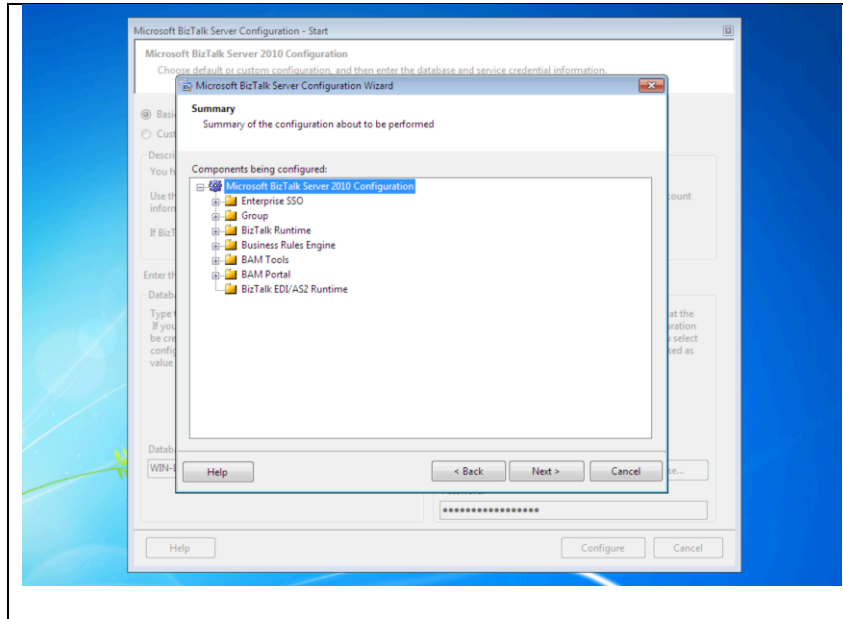
Figura CC : Biztalk Server, finalización de la instalación



Fuente: Programa de instalación de Biztalk Server

- Al finalizar la instalación de Biztalk Server se debe configurar el servidor, para ello se ejecuta la aplicación Biztalk Server Configuration, ingresamos el nombre de usuario y la contraseña del usuario o usuarios que van a manejar la aplicación (los usuarios deben también estar registrados en el manejo de usuarios de Windows). Al aceptar la configuración se debe observar en la pantalla resumen que se presenta a continuación que no existan advertencias acerca de los componentes y con esto se termina la instalación de Biztalk Server

Figura DD : Biztalk Server, resumen de la configuración del servidor



Fuente: Programa de instalación de Biztalk Server

3.4.2. Instalación de Oracle Fusion Middleware® SOA Suite.

El objetivo de la instalación de Oracle Fusion Middleware SOA Suite es crear una estructura definida por Oracle para su funcionamiento, esta estructura se encuentra detallada en la figura EE.

Figura EE : Estructura de Oracle Fusion Middleware SOA Suite

| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

- El primer paso de la instalación es crear el directorio c:\stageFMW en donde se van a colocar todos los instaladores necesarios.
- Se instalará la última versión del JDK de Java.

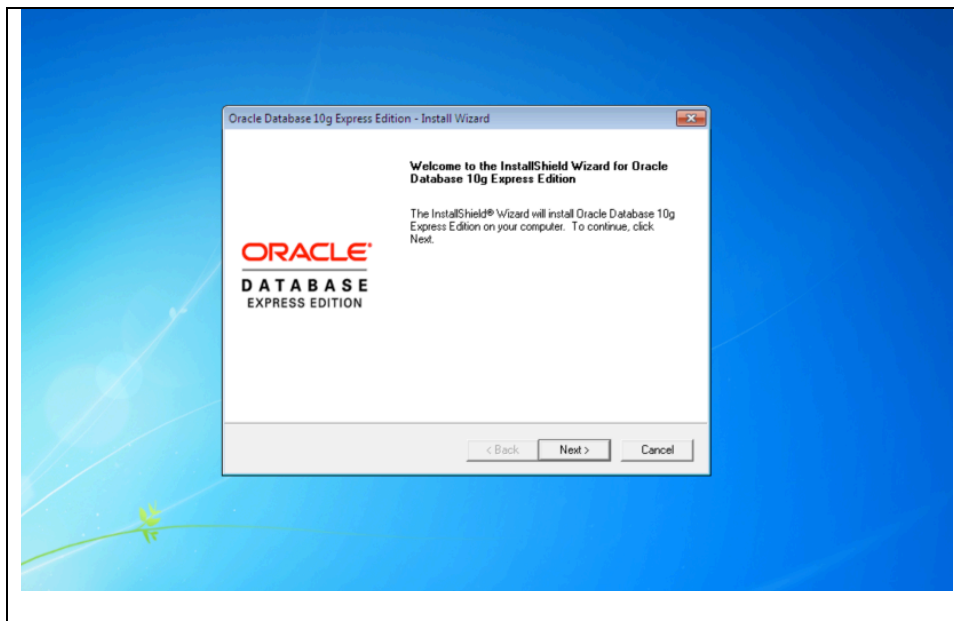
Figura FF : Instalación de Base de datos Oracle

| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

- Se iniciará con la instalación de la Base de datos, pueden ser las versiones completas o la edición Express, para el desarrollo de la investigación la base de datos escogida es la 10g Express Edition. En la instalación no se deberá hacer una configuración crítica, tan solo ingresar la clave del usuario administrador.

Figura GG : Instalación de Oracle Database 10g Express



Fuente: Programa de instalación de Oracle Fusion Middleware

- Finalizada la instalación de la base de datos, se debe configurar el parámetro de procesos a 300, para ello se irá a la línea de comandos de SQL y se ingresará el parámetro con la siguiente configuración:

```
SQL> alter system reset sessions scope=spfile sid='*';
SQL> alter system set processes=300 scope=spfile;
```

- El siguiente paso a seguir es la instalación de WebLogic Server, que es el servidor de aplicaciones de nuestra instalación de Oracle SOA.

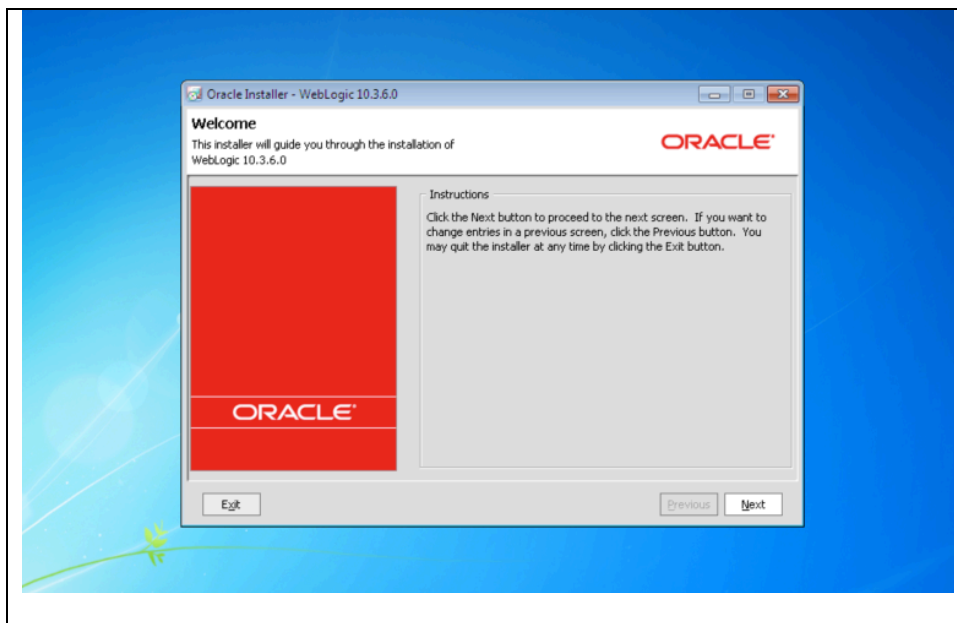
Figura HH : Instalación de WebLogic Server

| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

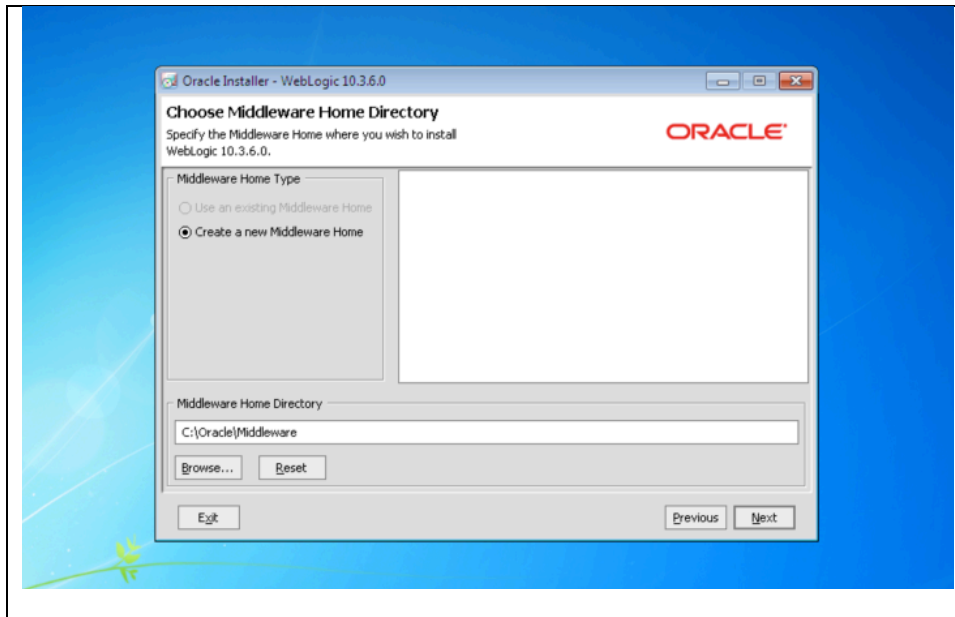
- En la instalación de WebLogic Server no se debe realizar ninguna configuración, tan solo seleccionar el lugar de la instalación y asegurarse que se hayan escogido todas las opciones de la instalación personalizada.

Figura II : Instalación de WebLogic Server



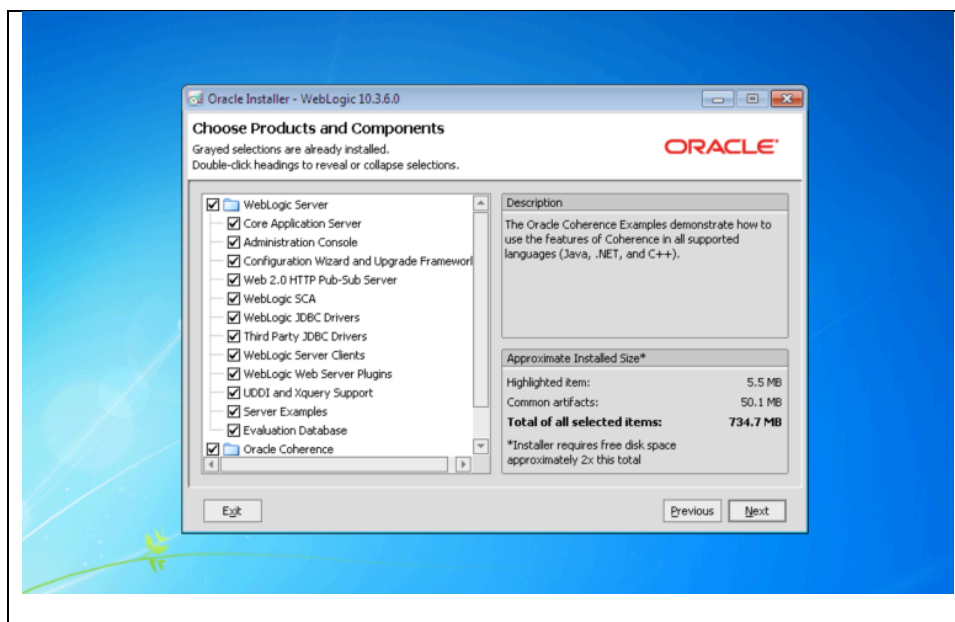
Fuente: Programa de instalación de Oracle Fusion Midleware

Figura JJ : Elección de la ruta de WebLogic Server



Fuente: Programa de instalación de Oracle Fusion Midleware

Figura KK : Elección de opciones de instalación de WebLogic Server



Fuente: Programa de instalación de Oracle Fusion Midleware

- El siguiente paso de la instalación es crear la estructura (Schema) necesaria en la base de datos para el funcionamiento de Oracle Fusion Middleware SOA Suite, para ello Oracle ha creado un instalador (Repository Creation Utility) que realizará esta tarea.

Figura LL : Creación de la estructura de la base de datos

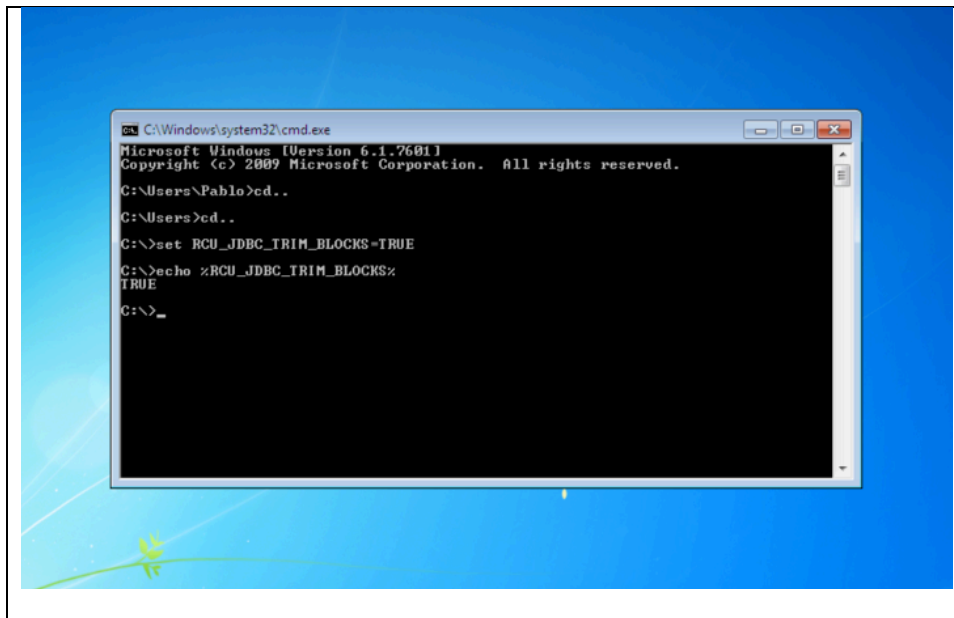
| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

- Antes de iniciar el instalador debemos cambiar los parámetros de 2 variables de ambiente en una línea de comandos de windows, únicamente si se está usando la versión Express (como es el caso de nuestra investigación).

```
set RCU_JDBC_TRIM_BLOCKS=TRUE
```

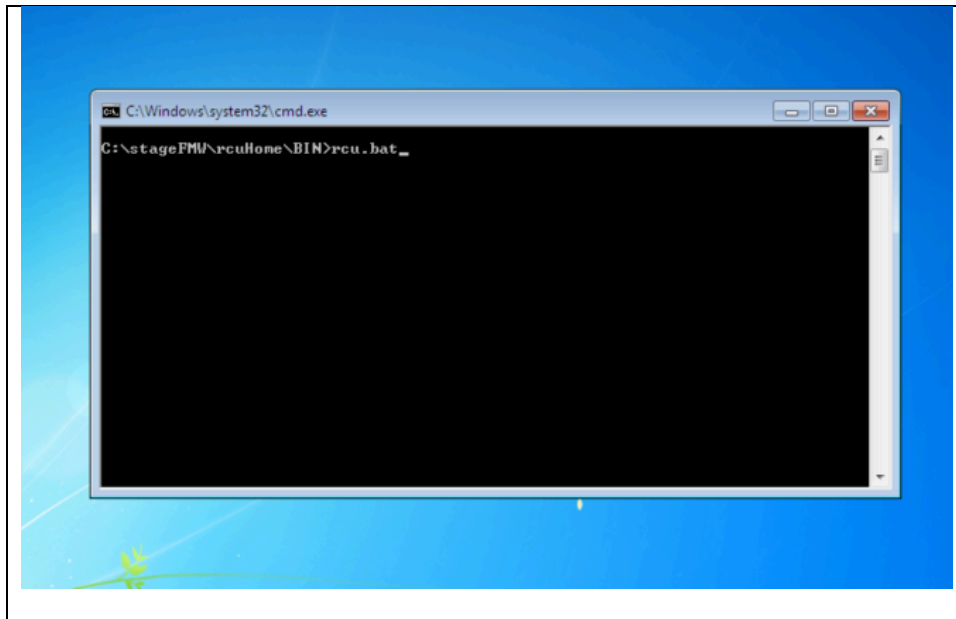
Figura MM : Configuración de las variables de ambiente para la aplicación



Autor: Juan Pablo Almeida Durán

- El inicio del instalador de Repository Creation Utility lo debemos hacer desde la línea de comandos de Windows ingresando al directorio c:\stageFMW\rcuHome\BIN y ejecutando el archivo rcu.bat

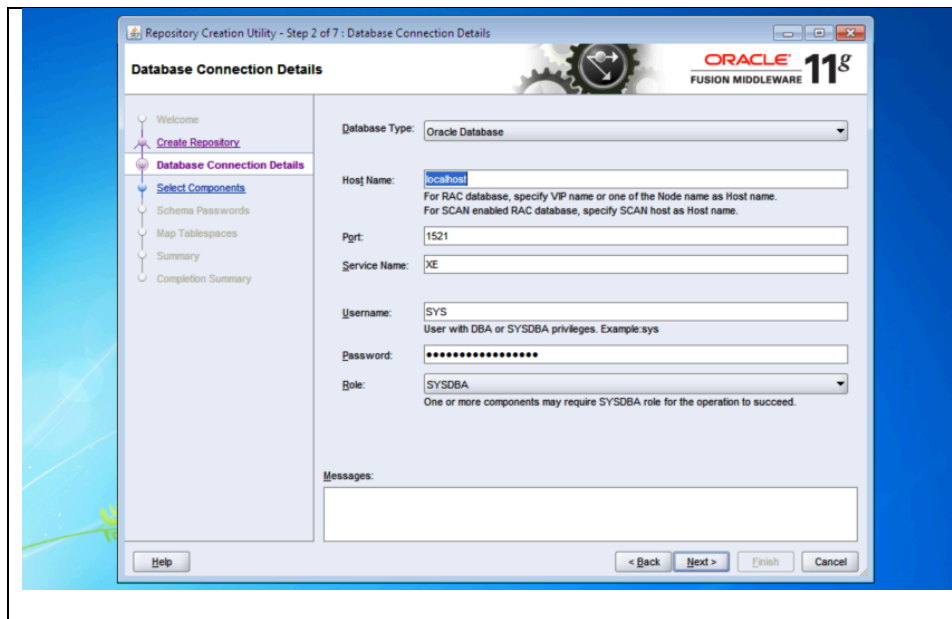
Figura NN : Inicio del instalador de Repository Creation Utility



Autor: Juan Pablo Almeida Durán

- Al iniciar el instalador navegaremos por diferentes pantallas, la primera únicamente preguntará si la operación a realizar es crear la estructura de la base de datos o eliminarla.
- En el segundo paso se debe ingresar la información de la base de datos creada en el paso 'Base de Datos'. La información requerida es el tipo de base de datos (Oracle), el nombre del servidor en el que se encuentra la base de datos (localhost en nuestro caso), el puerto de acceso a la base de datos (1521 por defecto), el nombre del servicio de la base de datos (XE nuestra base de datos Express), el usuario administrador de la base de datos y la clave creada en el paso 'Base de Datos'.

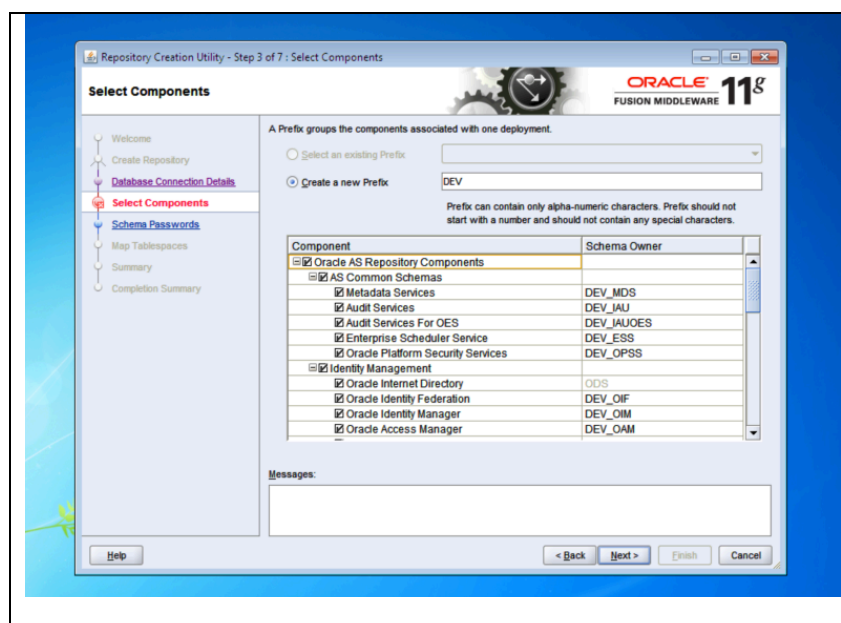
Figura OO : Ingreso de parámetros en Repository Creation Utility



Fuente: Programa de instalación de Oracle Fusion Midleware

- En este paso, al usar una base de datos Express, aparecerán ventanas de advertencia que en la versión completa de la base de datos no aparecerán, por lo que podemos ignorarlas.
- Luego se debe escoger las características a usar en la estructura que se va a crear, para la investigación se escogió todas las características disponibles.

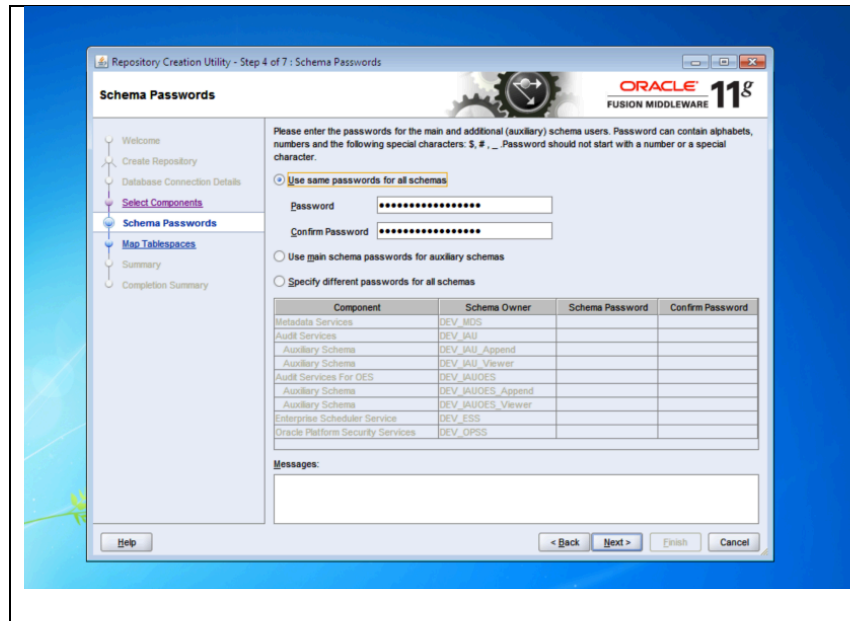
Figura PP : Elección de las características de Repository Creation Utility



Fuente: Programa de instalación de Oracle Fusion Midleware

- Una vez instaladas las características y componentes se debe ingresar la clave que tendrá esta estructura de tablas creada.

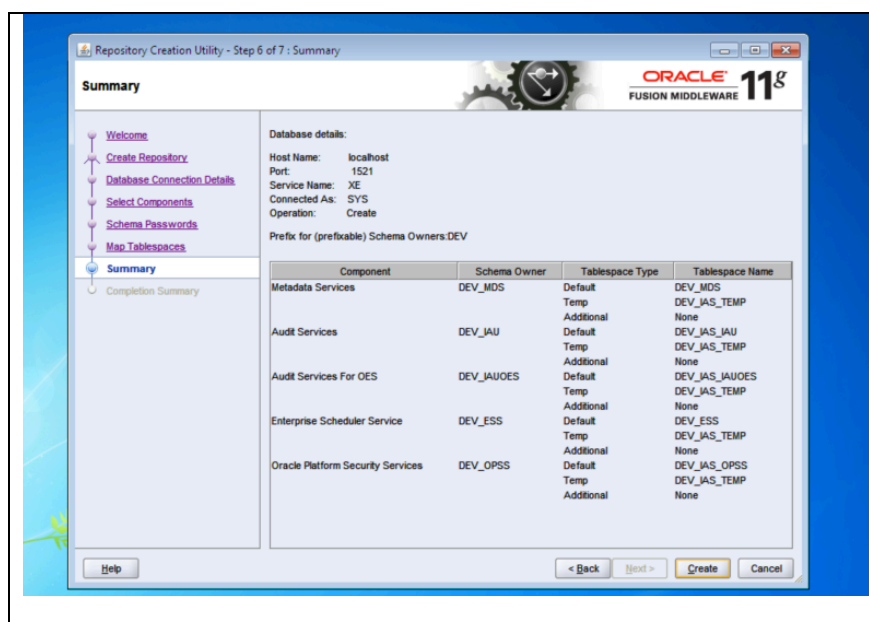
Figura QQ : Ingreso de la clave para los componentes de Repository Creation Utility



Fuente: Programa de instalación de Oracle Fusion Midleware

- Se mostrará una pantalla con las opciones escogidas y al presionar el botón 'Create' se creará la estructura requerida, este paso toma aproximadamente 2 minutos.

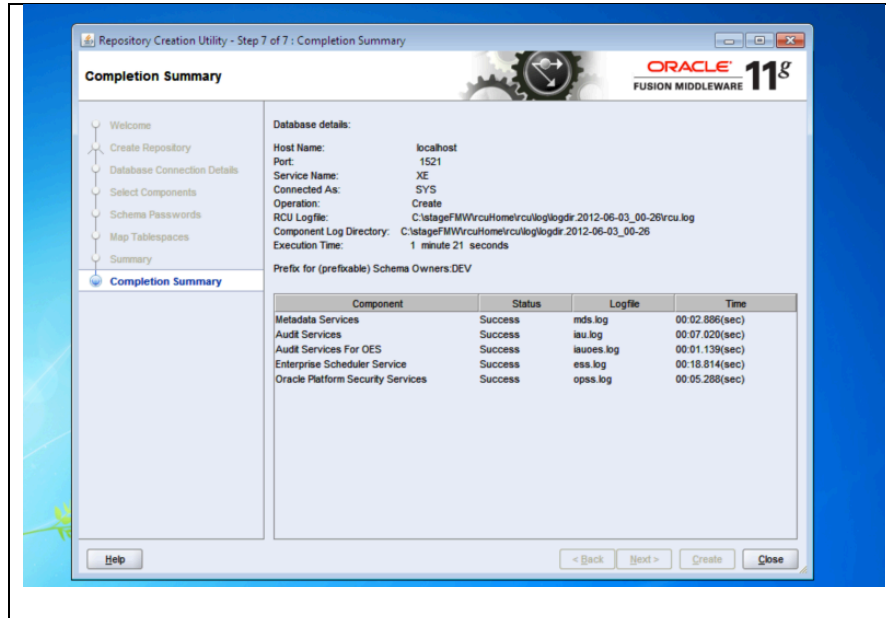
Figura RR : Creación de la estructura de Repository Creation Utility



Fuente: Programa de instalación de Oracle Fusion Midleware

- Al finalizar de crearse las tablas y el esquema de la base de datos necesaria para Oracle Fusion Middleware se nos presentará el resumen final en el que analizaremos si la instalación fue exitosa.

Figura SS : Finalización de la creación del esquema de tablas



Fuente: Programa de instalación de Oracle Fusion Midleware

- Realizada la creación del esquema de la base de datos es hora de crear el servidor SOA.

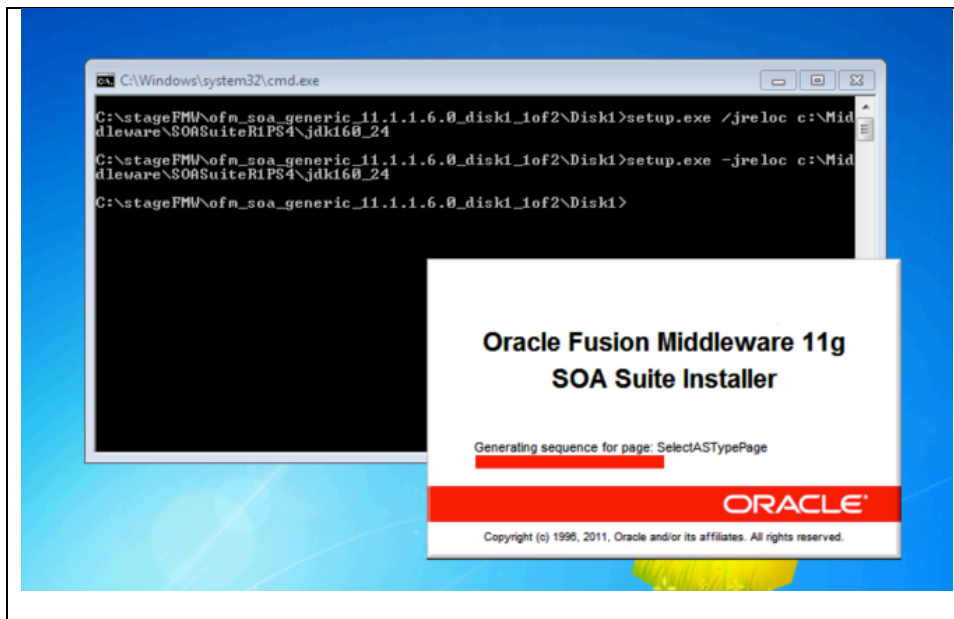
| | | | |
|-----|--------------------|-----------------|---------------|
| | Base de Datos | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

- Para la creación del servidor SOA debemos iniciar el instalador desde la línea de comandos de Windows, enviando como parámetro la ruta al JDK de Java.

```
C:\stageFMW\ofm_soa_generic_11.1.1.6.0_disk1_1of2\Disk1>setup.exe -jreloc
c:\Middleware\SOASuiteR1PS4\jdk160_24
```

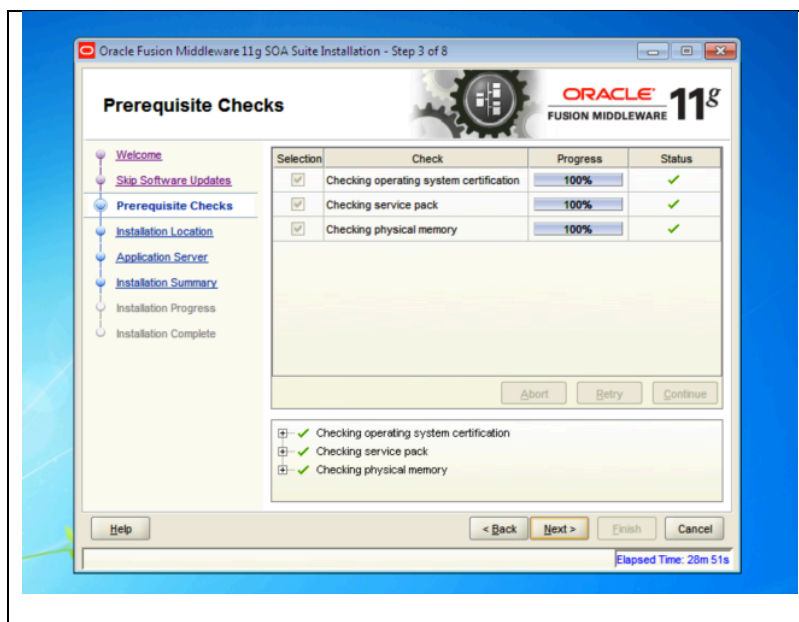
Figura TT : Instalación del servidor SOA



Fuente: Programa de instalación de Oracle Fusion Midleware

- En la primera pantalla se preguntará si se desea actualizar el software, es recomendable actualizarlo.
- El instalador revisará si todos los pre-requisitos anteriores se encuentran correctamente instalados.

Figura UU : Pre Requisitos del Servidor SOA



Fuente: Programa de instalación de Oracle Fusion Midleware

- El programa de instalación reconocerá automáticamente el servidor de aplicaciones si es Weblogic Server o WebSphere de IBM, caso contrario el usuario deberá localizar el lugar donde se encuentra instalado.

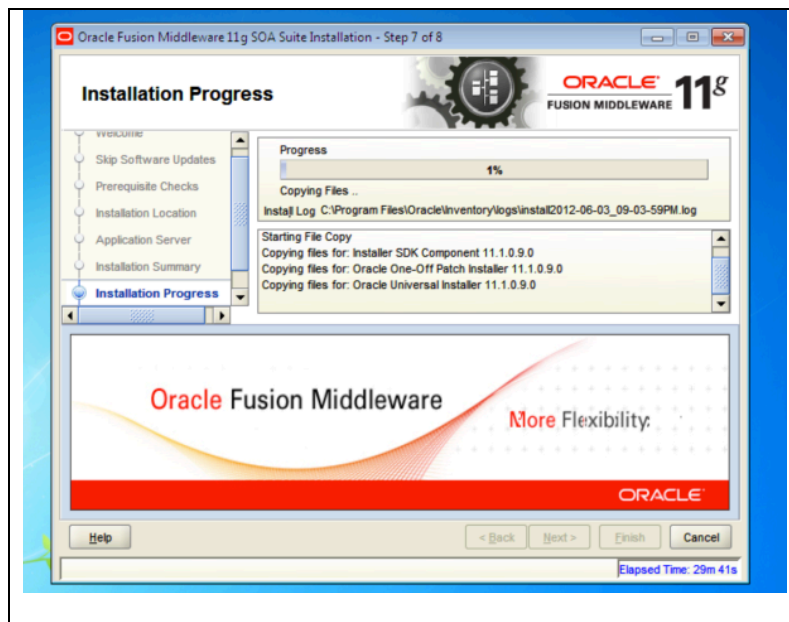
Figura VV : Ubicación de SOA Server



Fuente: Programa de instalación de Oracle Fusion Midleware

- Se iniciará la instalación del servidor SOA.

Figura WW : Progreso de la instalación del servidor SOA



Fuente: Programa de instalación de Oracle Fusion Midleware

Figura XX : Instalación del Bus de Servicio Empresarial Oracle

| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

- El bus de servicio de Oracle posee un inicio de instalación similar al servidor SOA, en donde se debe iniciar el instalador por la línea de comando de Windows enviando como parámetro la ubicación del JDK de java.

```
C:\stageFMW\ofm_osb_generic_11.1.1.6.0_disk1_1of1\Disk1>setup.exe -jreloc
c:\Middleware\SOASuiteR1PS4\jdk160_24
```

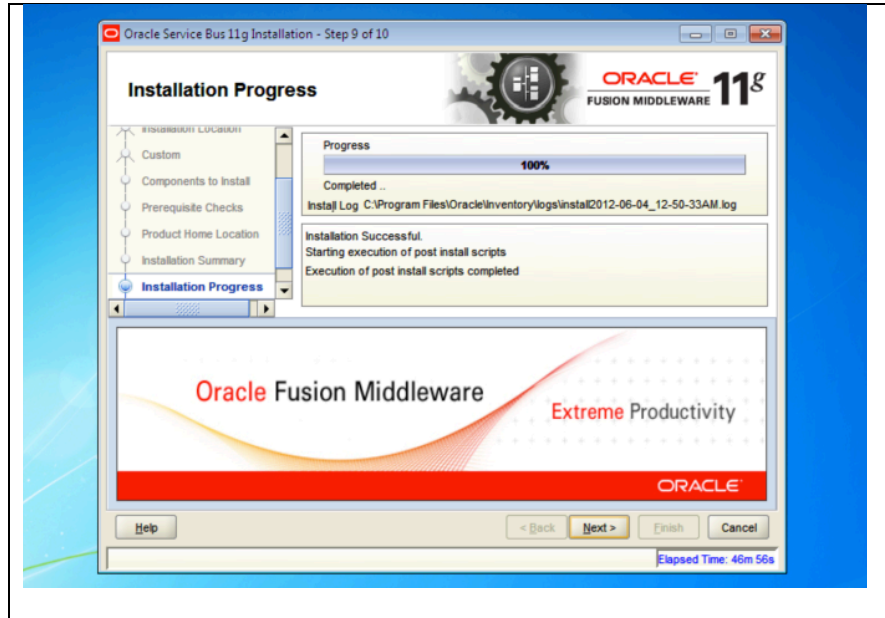
Figura YY : Instalación del Bus de Servicio de Oracle



Fuente: Programa de instalación de Oracle Fusion Midleware

- El instalador reconocerá automáticamente la ubicación del servidor de aplicaciones Weblogic, y el sitio de instalación del bus de servicio de Oracle, con lo cual empezará a instalar los componentes.

Figura ZZ : Finalización de la instalación del Bus de Servicio de Oracle



Fuente: Programa de instalación de Oracle Fusion Midleware

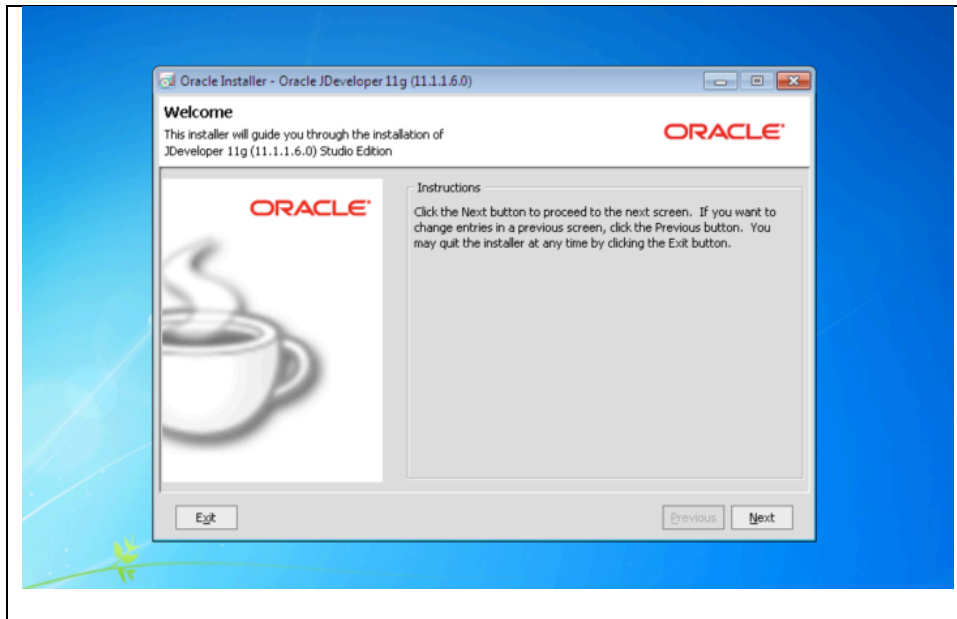
Figura AAA : Instalación de JDeveloper y extensiones SOA

| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

- La instalación de JDeveloper se la realizará directamente desde el instalador proporcionado por Oracle (jdevstudio<versión>.exe) con permisos de administrador de Windows.

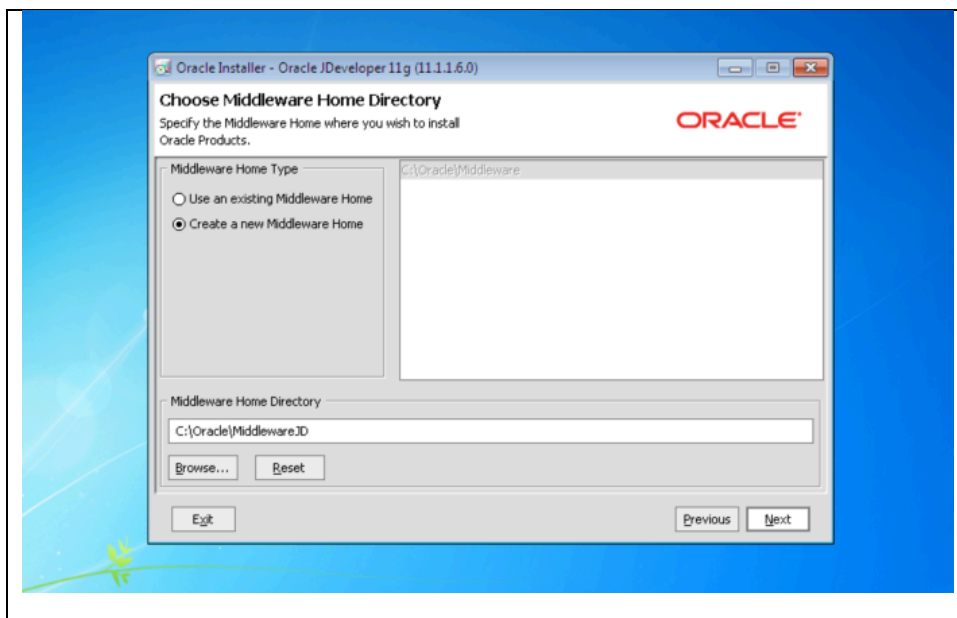
Figura BBB : Instalación de JDeveloper



Fuente: Programa de instalación de Oracle Fusion Middleware

- El siguiente paso requerido es la ubicación en la cual se instalará JDeveloper. Se debe escoger una ubicación que difiera de la carpeta C:\Oracle\Middleware ya que presentará conflictos.

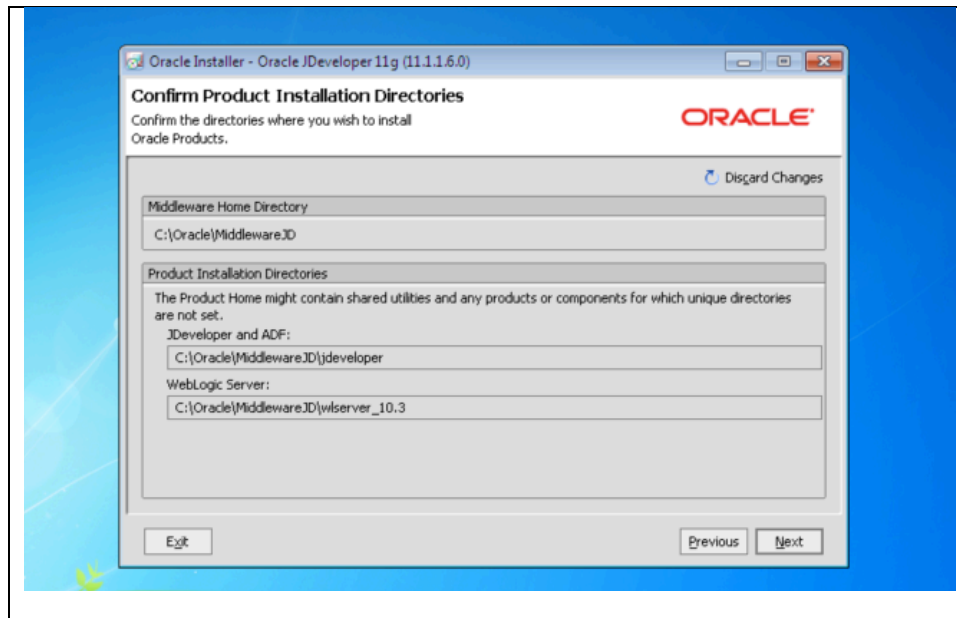
Figura CCC : Elección de la ubicación de JDeveloper



Fuente: Programa de instalación de Oracle Fusion Middleware

- Escogemos el tipo de instalación que deseamos (Se recomienda la instalación completa. Se presentará un resumen con las opciones que se instalarán.

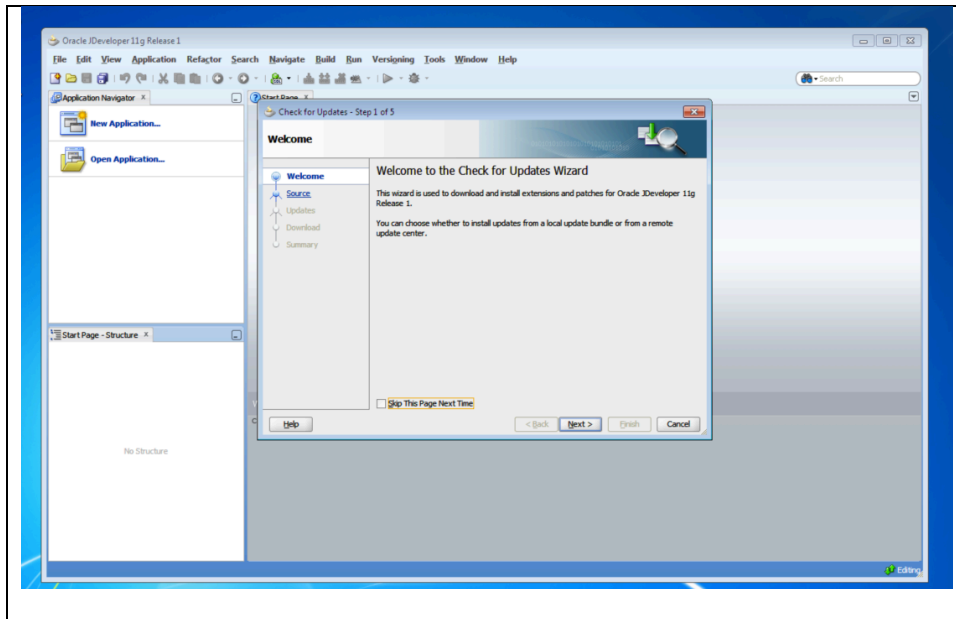
Figura DDD : Opciones a instalar de JDeveloper



Fuente: Programa de instalación de Oracle Fusion Midleware

- Al finalizar la instalación de JDeveloper iniciará la aplicación desde la cual se instalarán las extensiones SOA para finalizar la instalación de Oracle Fusion Middleware SOA Suite.

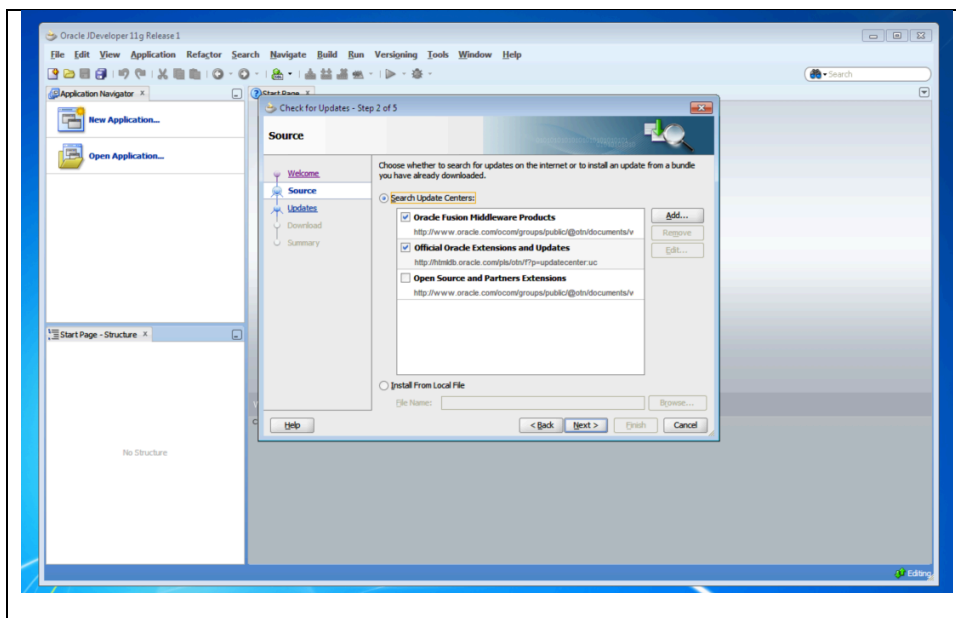
Figura EEE : Actualización de extensiones SOA



Fuente: Programa de instalación de Oracle Fusion Midleware

- Escogemos los servidores donde JDeveloper buscará las actualizaciones (Oracle Fusion Middleware Products y Official Oracle Extensions and Updates).

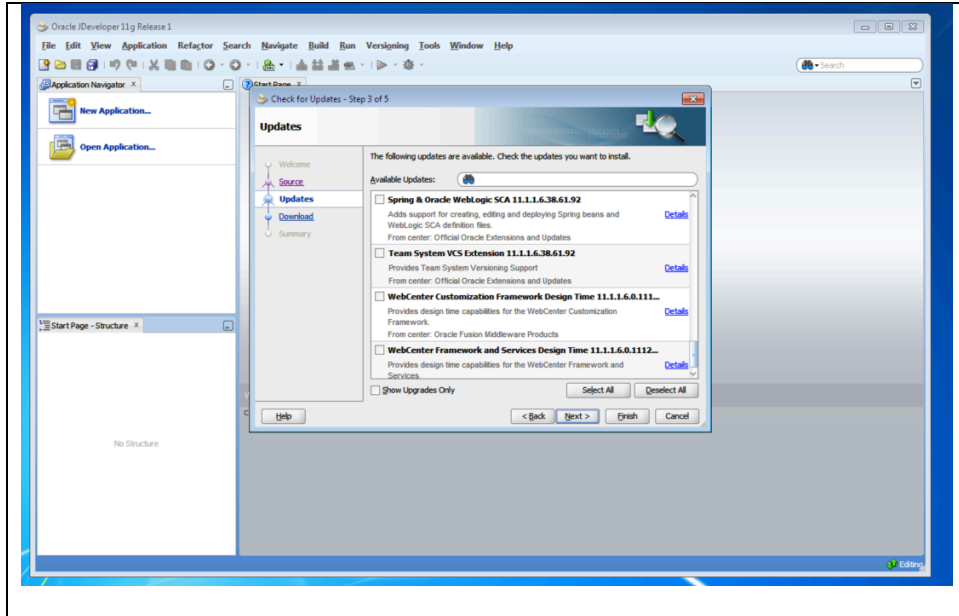
Figura FFF : Servidores de actualización de extensiones SOA



Fuente: Programa de instalación de Oracle Fusion Midleware

- Se escoge las 2 extensiones requeridas para Oracle Fusion Middleware SOA Suite. (Oracle BPM Studio y Oracle SOA Composite Studio).

Figura GGG : Componentes a actualizar de JDeveloper



Fuente: Programa de instalación de Oracle Fusion Midleware

- Al finalizar la actualización de extensiones SOA para JDeveloper se ha completado la estructura necesaria para que Oracle Fusion Middleware funcione como herramienta de integración de aplicaciones bajo la arquitectura SOA.

Figura HHH : Finalización de la instalación de Oracle Fusion Middleware SOA Suite

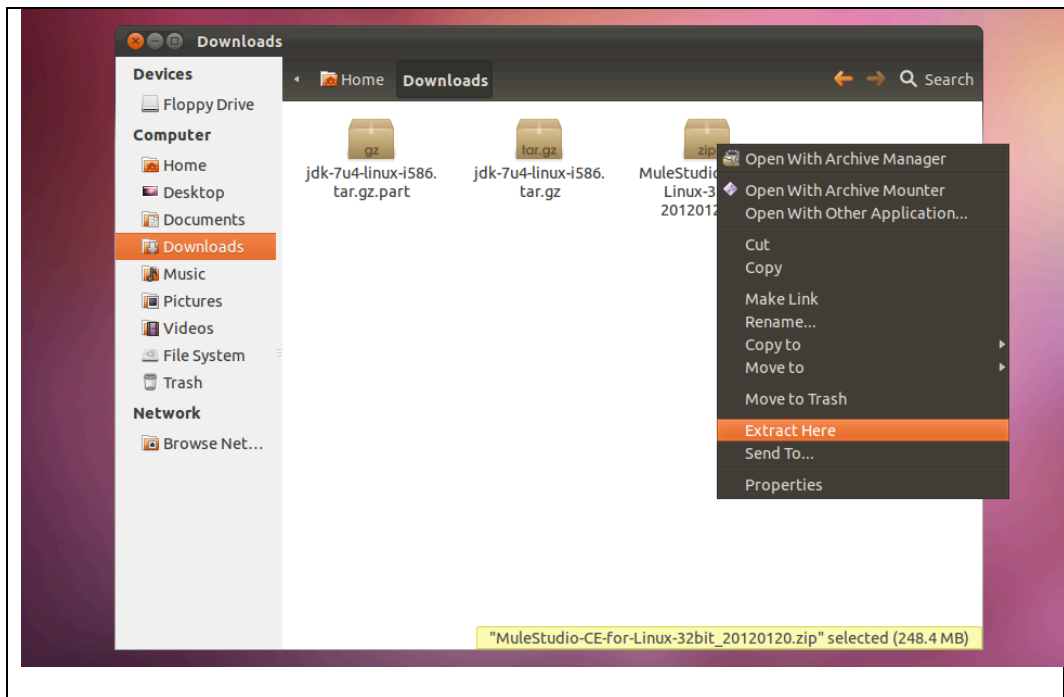
| | | | |
|---------------|--------------------|-----------------|---------------|
| Base de Datos | | WebLogic Server | RCU |
| SOA | Oracle Service Bus | JDeveloper | SOA Extension |

Autor: Juan Pablo Almeida Durán

3.4.3. Instalación de Mule ESB® Community Edition

- Al descargar Mule ESB Community Edition de la página <http://www.mulesoft.org/download-mule-esb-community-edition> procedemos a extraer el archivo .zip.

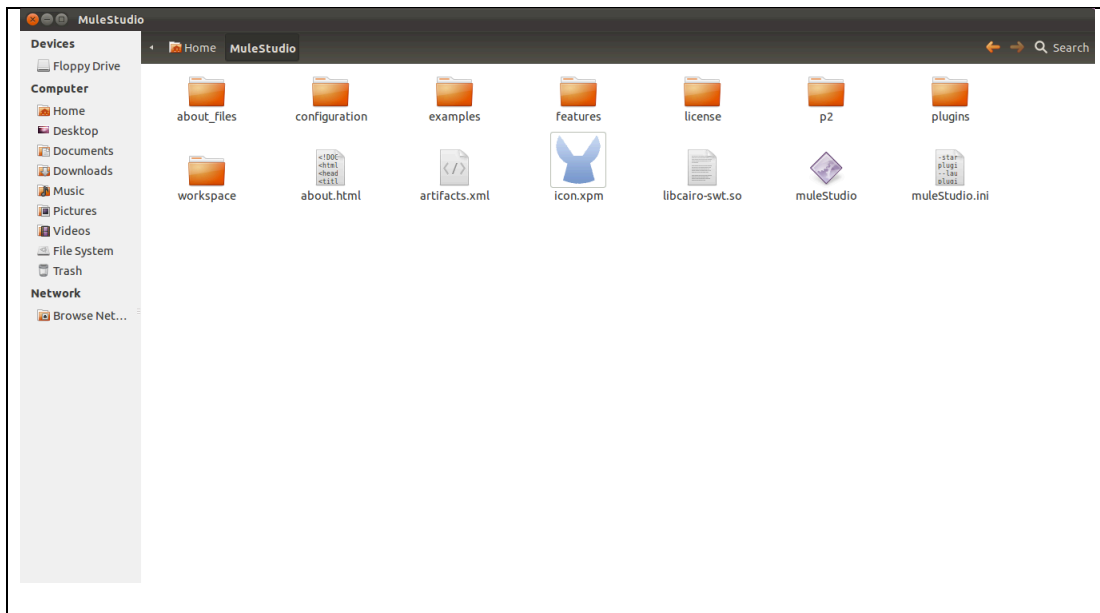
Figura III : Extracción de Mule ESB® Community Edition



Fuente: Programa de instalación de Mule ESB Community Edition

- Al extraer el archivo obtendremos un subdirectorio llamado /MuleStudio
- Se moverá este subdirectorio al lugar donde se desee que esté instalado Mule Studio, la única restricción que existe es que el directorio no contenga espacios en blanco.
- Dentro de la carpeta /MuleStudio se encuentra el archivo MuleStudio con el cual ejecutamos el software.

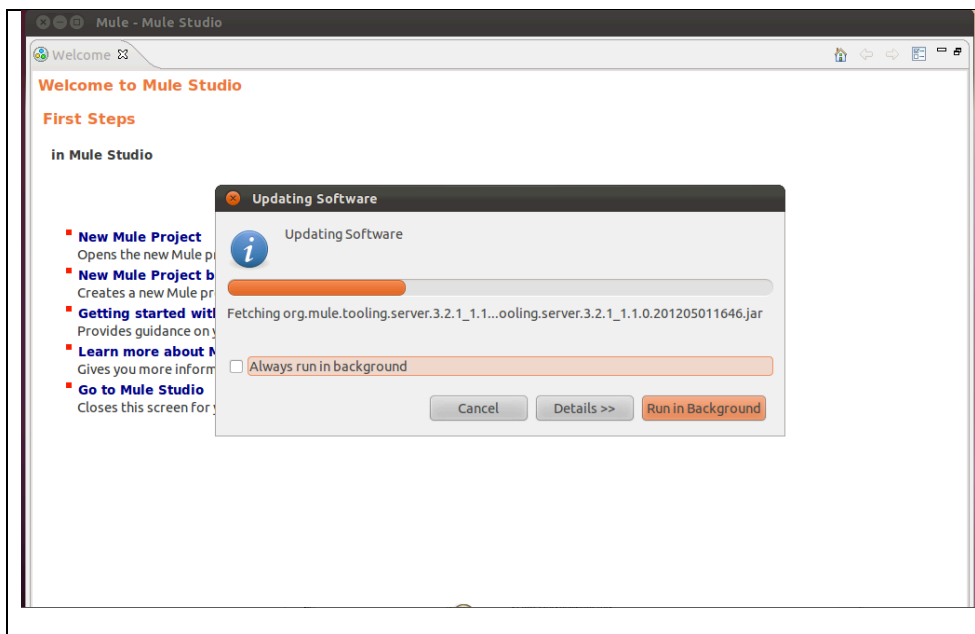
Figura JJJ : Contenido del directorio de Mule ESB Community Edition



Fuente: Programa de instalación de Mule ESB Community Edition

- Cuando se inicie por vez primera Mule ESB Community Edition requerirá el permiso para realizar actualizaciones.

Figura KKK : Actualización de Mule ESB Community Edition



Fuente: Programa de instalación de Mule ESB Community Edition

3.5. Análisis del proceso de integración de aplicaciones utilizando las herramientas escogidas para interfaces entre aplicaciones.

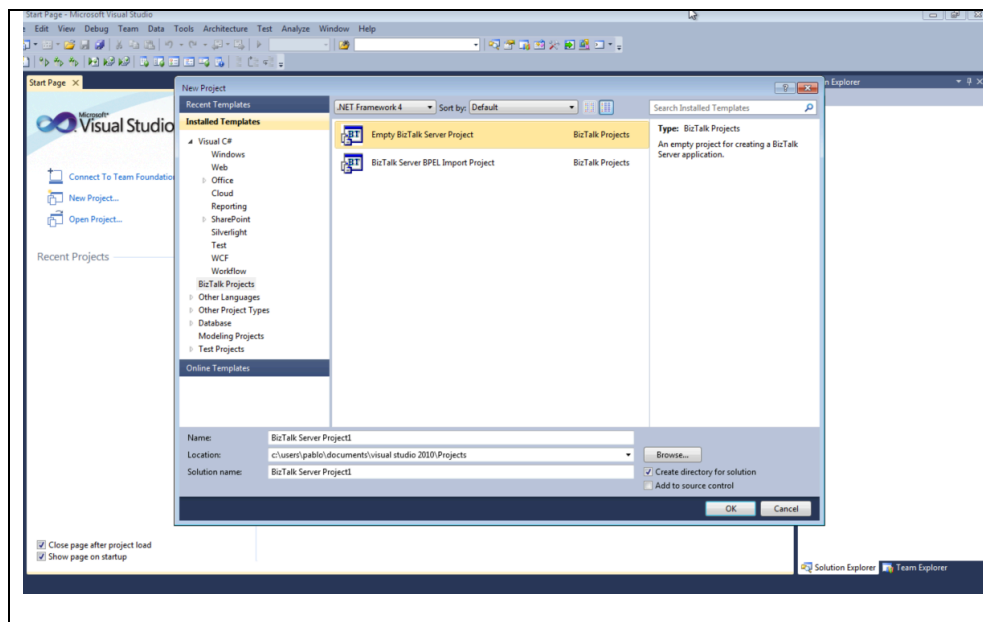
3.5.1. Integración de aplicaciones utilizando Microsoft® Biztalk® Server.

El escenario de pruebas de integración de Microsoft Biztalk Server constará de un servicio que recibe una petición acerca de la actualización de estados de un usuario en la red social Twitter para luego recibir una respuesta del estado y enviarlo a otro servicio web que puede ser de mensajería por correo electrónico.

El software para desarrollar esta prueba será Visual Studio 2010 enterprise edition, versión de evaluación.

- Se creará un proyecto nuevo de Biztalk Server en Visual Studio en el menú New, Empty Biztalk Server Project.

Figura LLL : Creación de un proyecto de Biztalk Server



Fuente: Microsoft Biztalk Server

- Creado el proyecto sobre el cual se trabajará y se definirá la estructura de los mensajes de peticiones y respuestas.
- Se crea el esquema del mensaje que realizará la petición a Twitter, para ello se ingresa al menú Añadir, nuevo ítem, y se escoge Schema Files.

Para crear la estructura del mensaje se toma como referencia la tabla de petición de las librerías de Twitter que se encuentra en su página para desarrolladores <https://api.twitter.com/> .

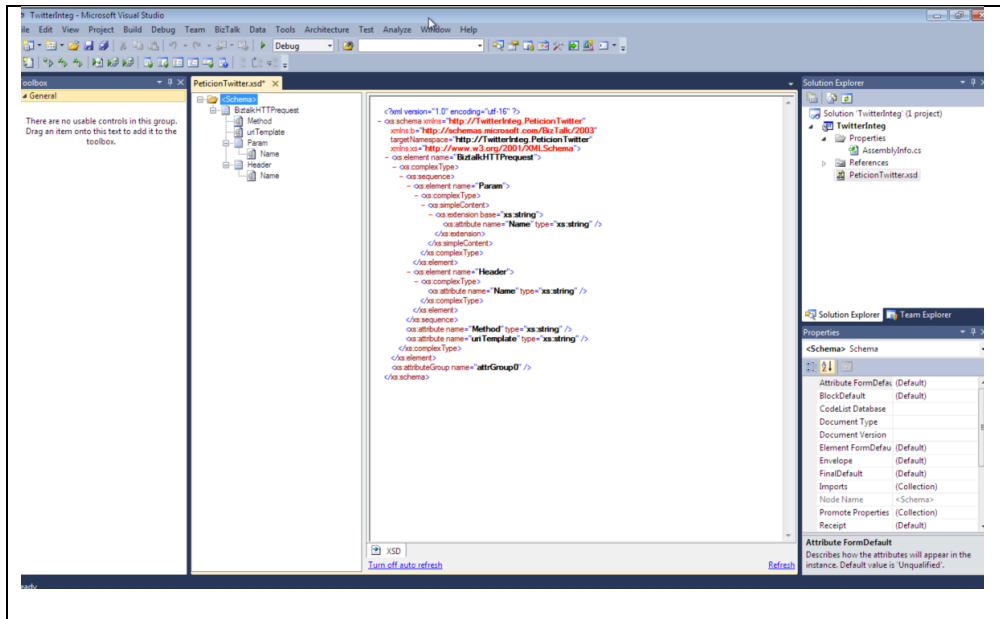
| Node name (Nombre del nodo) | Type (tipo de dato) | Property Name to be Set (propiedad) | Property Value (valor inicial) |
|--|--|--|---|
| <Schema> | - | Element FormDefault | Qualified |
| | | Target Namespace | http://microsoft.com/schemas/samples/biztalkwebhttp/1.0 |
| bizTalkWebHttpRequest | Root node | Data Structure Type | bizTalkWebHttpRequest |
| | | RootNode TypeName | bizTalkWebHttpRequest |
| method (The value for this attribute will specify the method to be invoked on the Twitter API) | Child attribute to bizTalkWebHttpRequest | Data Type | xs:string |
| uriTemplate (The value for this attribute will be used to frame the URL) | Child attribute to bizTalkWebHttpRequest | Data Type | xs:string |
| param | Child record under bizTalkWebHttpRequest | Data Structure | keyValue |

| | | | |
|--|--|---------------------|-----------|
| | uest | Type | |
| | | Max Occurs | unbounded |
| | | Min Occurs | 0 |
| | | Mixed | True |
| name (The value for this attribute will be used as parameters appended to the URL) | Child attribute to param | Data Type | xs:string |
| header | Child record under bizTalkWebHttpRequest | Data Structure Type | keyValue |
| | | Max Occurs | unbounded |
| | | Min Occurs | 0 |
| | | Mixed | True |
| name (The value for this attribute will be used to specify the encoding in the message header that will be sent to invoke the Twitter API) | Child attribute to header | Data Type | xs:string |

Tabla 22 : Propiedades de un mensaje de petición de estados a Twitter⁵⁴

⁵⁴ Tomado de <https://api.twitter.com/>

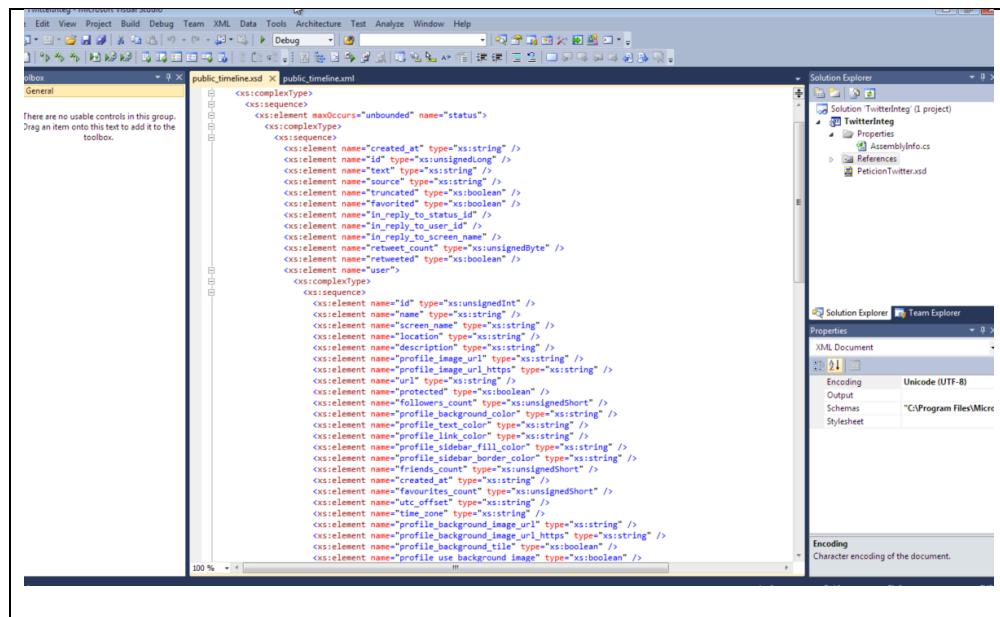
Figura MMM : Resultado de las configuraciones de petición de estados



Fuente: Microsoft Biztalk Server

- Se descargará el archivo public_timeline.xsd de la página de desarrolladores de Twitter y se lo añadirá al proyecto de Visual Studio, este archivo servirá para el mapeo de tipos de datos para la petición de estados de usuarios en Twitter.

Figura NNN : Archivo public_timeline.xsd



Fuente: Microsoft Biztalk Server

- Otro componente importante para esta integración es el esquema del mensaje que podrá ser usado posteriormente para enviar el estado de un usuario por correo electrónico, por lo que debemos tener un esquema de mensaje que reciba la respuesta de Twitter. Para la creación de este esquema, debemos ir a Añadir, nuevo ítem, Schema Files. Al igual que para el mensaje de petición a Twitter debemos seguir la estructura para un mensaje de una tabla de configuración.

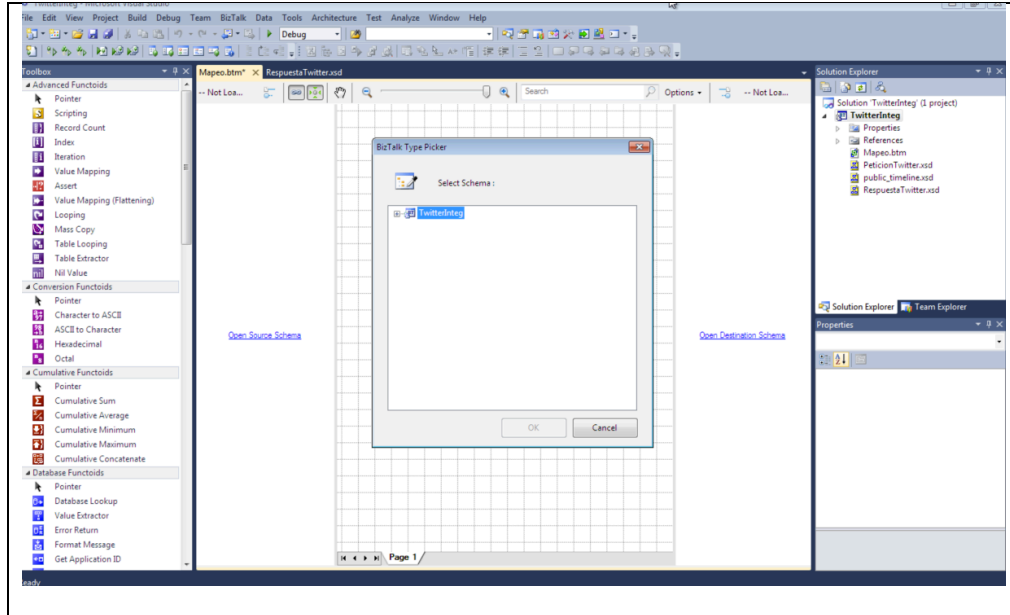
| Node Name (nombre del nodo) | Type (tipo de dato) | Property Name to be Set (propiedad) | Property Value (valor inicial) |
|--|--|--|---------------------------------------|
| <Schema> | | - | - |
| HTML | Root node | - | - |
| Body | Child record under the HTML element | - | - |
| P | Child field element under Body element | Data Type | xs:string |
| | | Max Occurs | unbounded |
| | | Min Occurs | 0 |

Tabla 23: Estructura de almacenamiento de un estado de Twitter⁵⁵

⁵⁵ Tomado de <https://api.twitter.com/>

en Open Destination Schema se escogerá el esquema de mensaje destinatario.

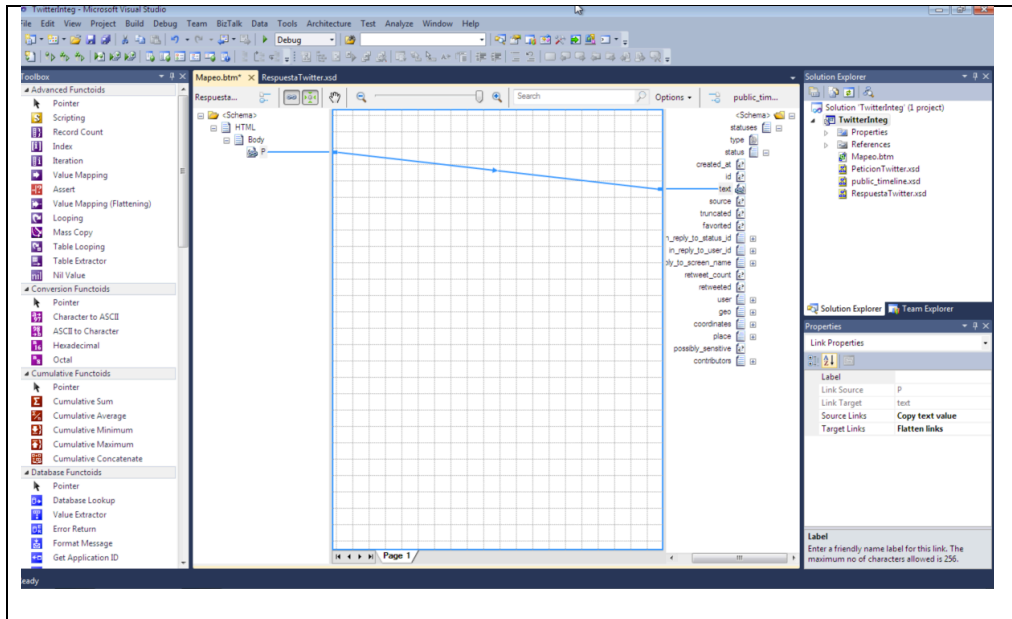
Figura QQQ : Elección de fuentes y destinos de datos



Fuente: Microsoft Biztalk Server

- Escogidas las fuentes y destinos de datos de la estructura de mensajes, se podrá señalar a que campo del mensaje receptor ira el campo del mensaje fuente, para ello solo se deberá escoger el campo fuente y direccionarlo al campo receptor, que serán del mismo tipo y quedará hecho el mapeo.

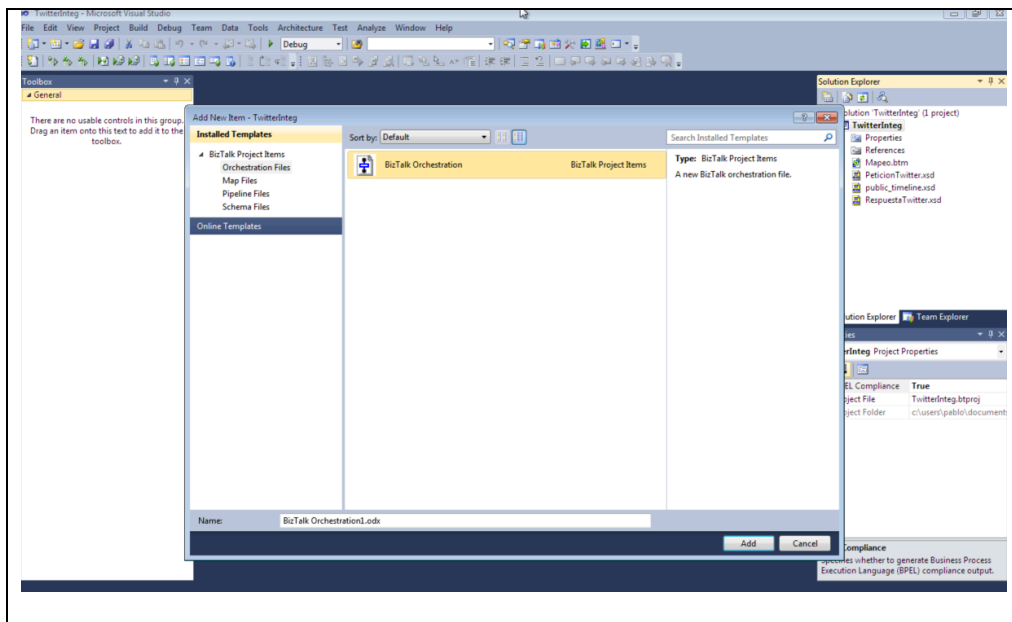
Figura RRR : Mapeo de datos



Fuente: Microsoft Biztalk Server

- Una vez realizado el mapeo de datos se debe realizar la orquestación de servicios en la cual se podrá ver el flujo de integración, para ello se debe presionar click derecho sobre el proyecto, ir al sub menú Add Item, Orchestration Files, Biztalk Orchestration.

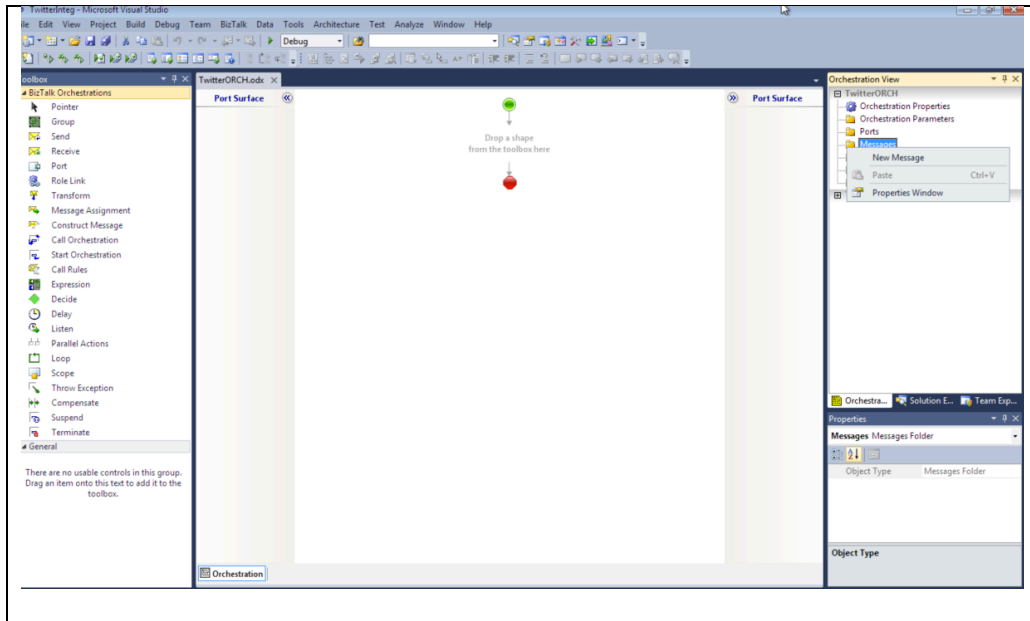
Figura SSS : Creación de la orquestación de servicios



Fuente: Microsoft Biztalk Server

- Añadimos 3 estructuras de mensajes en el menú de orquestación, estos mensajes serán Request, Response, Output.

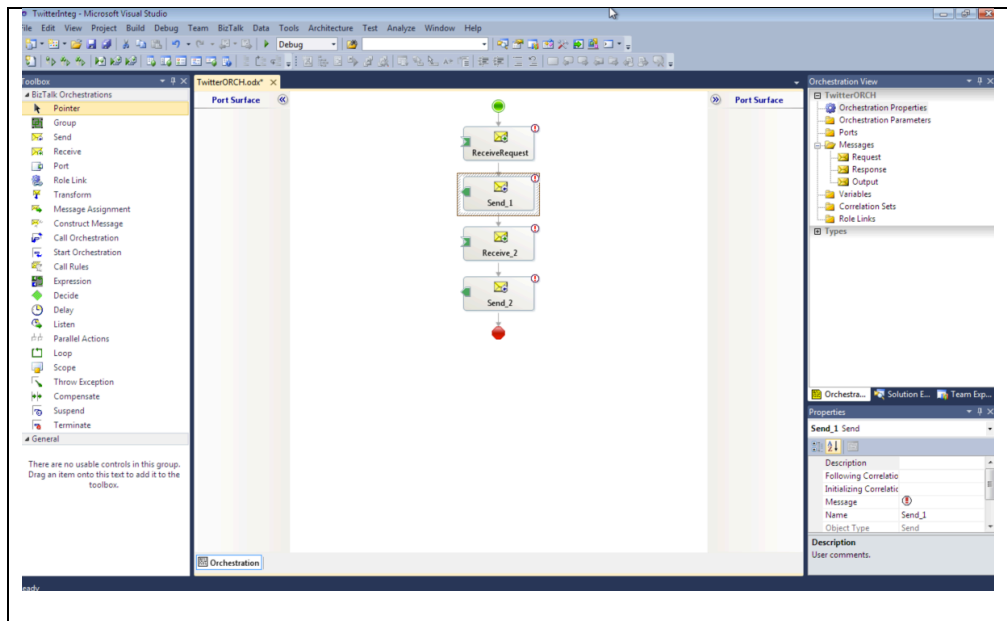
Figura TTT : Creación de la estructura de mensajes para la orquestación de servicios



Fuente: Microsoft Biztalk Server

- Se escoge 4 figuras de mensajes de la barra de herramientas de Biztalk Orchestration, 2 del tipo Send y dos del tipo Receive. El primer Mensaje Receive, recibirá la petición del servicio de un usuario y enviará esa petición por medio de un mensaje Send hacia Twitter para obtener la actualización del estado del usuario, luego recibirá la respuesta de Twiter en un mensaje Receive para enviarlo por el último mensaje Send por email.

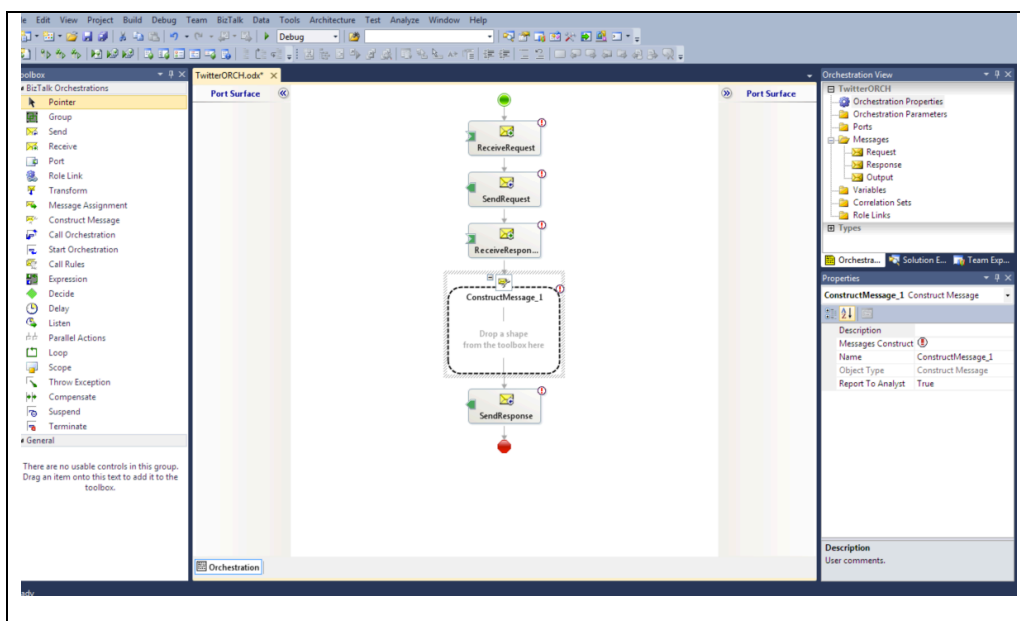
Figura UUU : Figuras de mensajes en el flujo



Fuente: Microsoft Biztalk Server

- Antes de terminar el flujo con el envío de la respuesta debemos construir la parte del mapeo que tomará la respuesta de Twitter y la empaquetará para enviarla por correo, para ello se insertará una función para construir el mensaje llamada ConstructMessage que la pondremos antes del último mensaje Send como en la figura.

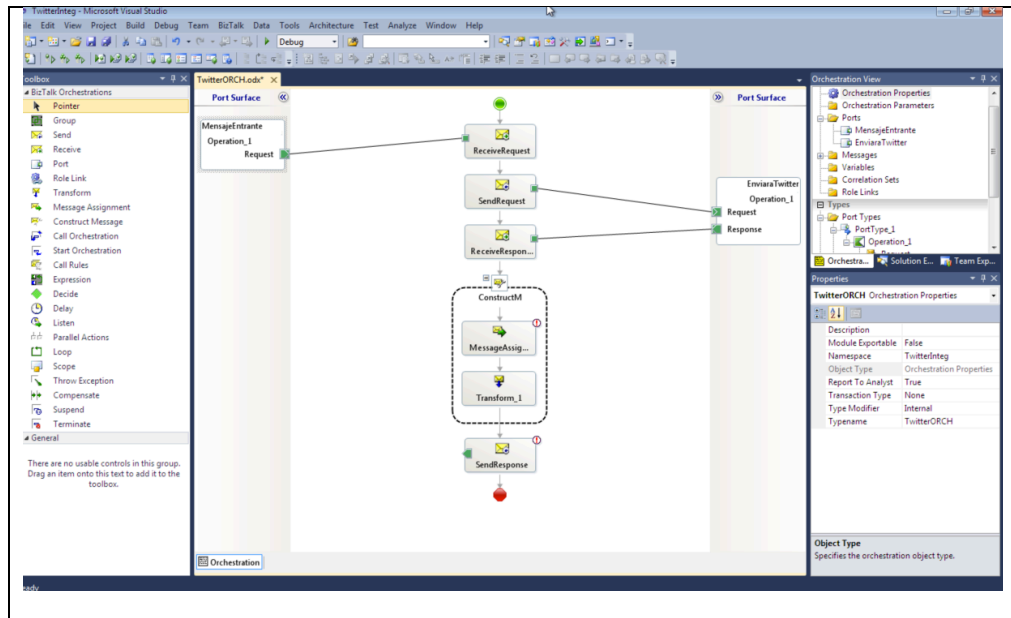
Figura VVV : Ingreso de ConstructMessage en el flujo



Fuente: Microsoft Biztalk Server

- Al ingresar en ConstructMessage escogemos las operaciones de transformación y asignación con lo cual tendremos la estructura final del flujo de integración de servicios web con los servicios expuestos de Twitter.

Figura WWW : Estructura final del flujo de integración



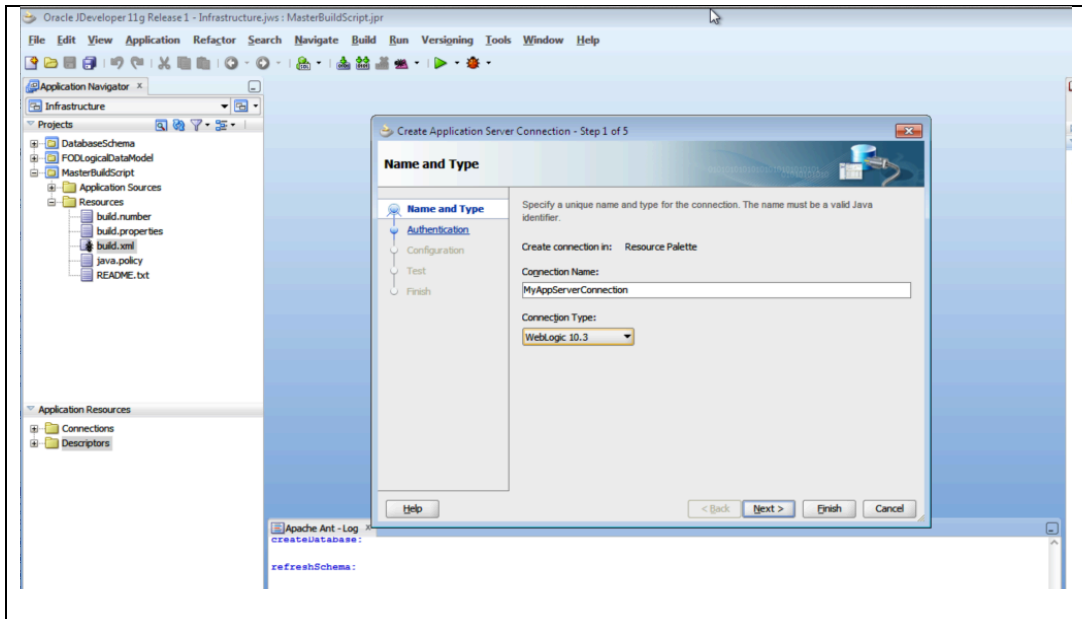
Fuente: Microsoft Biztalk Server

3.5.2. Integración de aplicaciones utilizando Oracle® Fusion® Middleware SOA Suite.

El escenario de integración de Oracle Fusion Middleware SOA Suite se realizó sobre 2 servicios web que fueron unidos mediante un proceso BPEL de integración para lo cual fue usado JDeveloper que es parte de la infraestructura de integración de Oracle Fusion Middleware.

- Al abrir JDeveloper se debe configurar el servidor de pruebas de aplicaciones, para ello se debe ingresar en el menú New, Server Connection.

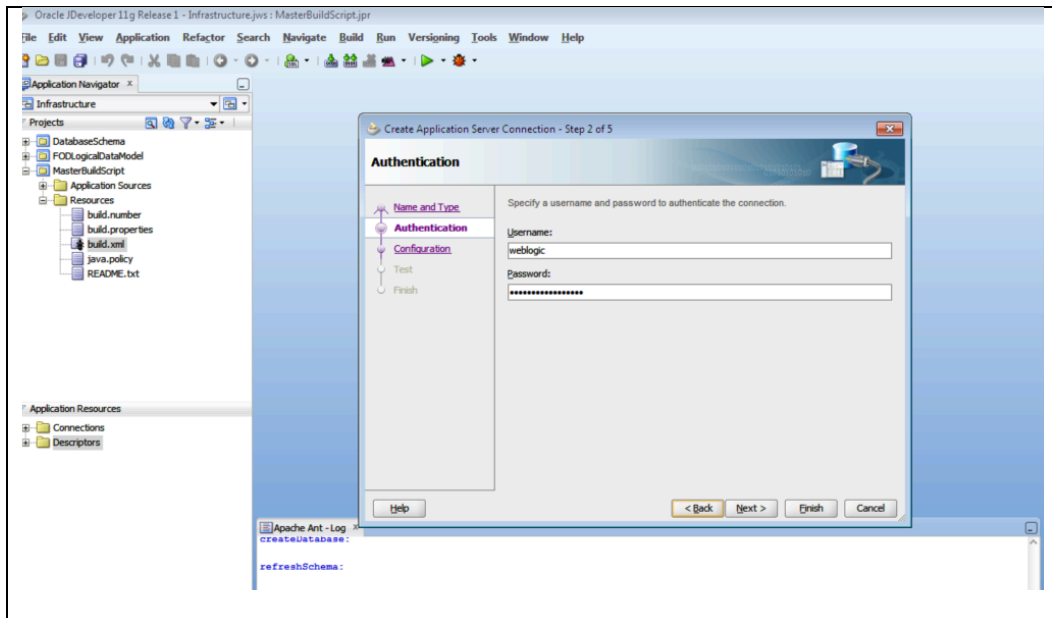
Figura XXX : Conexión al servidor con JDeveloper



Fuente: Oracle Fusion Middleware SOA Suite

- En la siguiente pantalla de la configuración serán requeridas las credenciales de weblogic server que fueron configuradas en la instalación del mismo.

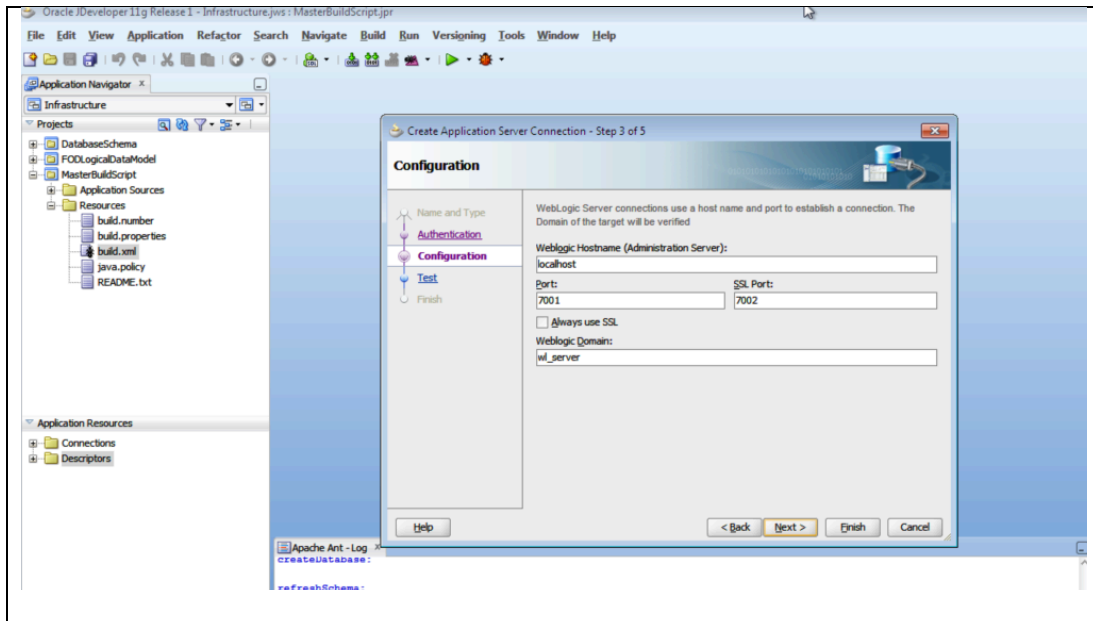
Figura YYY : Ingreso de credenciales de Weblogic Server



Fuente: Oracle Fusion Middleware SOA Suite

- Será requerida la confirmación del Hostname de Weblogic, el cual fue configurado por omisión como localhost y el puerto de escucha de peticiones.

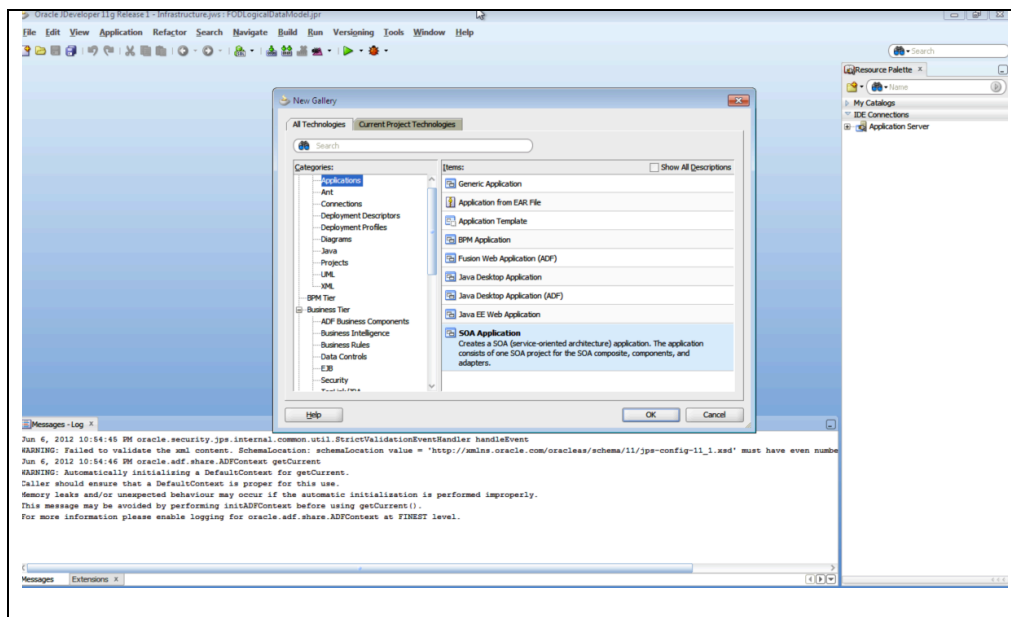
Figura ZZZ : Hostname de WebLogic



Fuente: Oracle Fusion Middleware SOA Suite

- En este instante se podrá crear la estructura de integración, para ello se ingresa al menú New, y se escoge la opción New Application

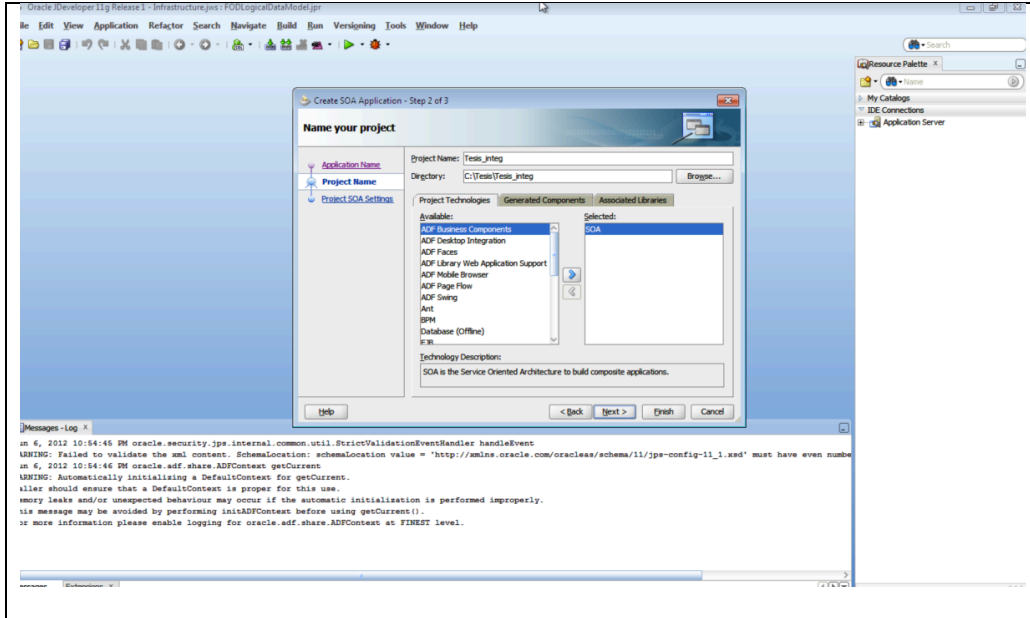
Figura AAAA : Creación de la infraestructura de integración



Fuente: Oracle Fusion Middleware SOA Suite

- Se ingresará el nombre del proceso de integración y se escogerá SOA en la barra de tecnologías del proyecto.

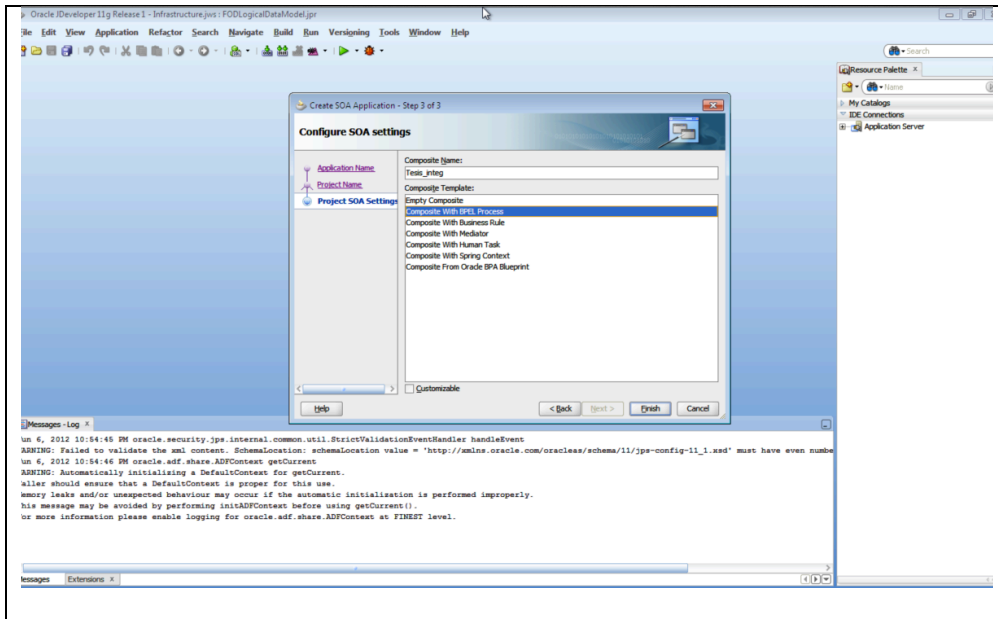
Figura BBBB : Elección de la tecnología de integración



Fuente: Oracle Fusion Middleware SOA Suite

- Se debe definir la plantilla de integración sobre la cual se trabajará en el espacio de trabajo, en el caso de la investigación se escogerá Composite with BPEL Process.

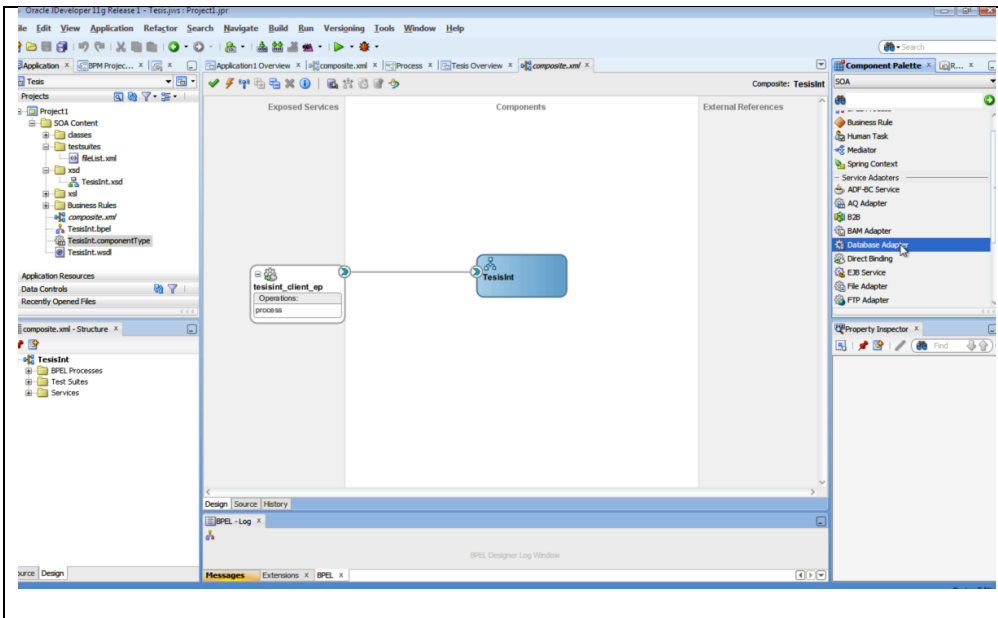
Figura CCC : Plantilla de integración



Fuente: Oracle Fusion Middleware SOA Suite

- En el espacio de trabajo de JDeveloper se presentará la estructura inicial de integración de acuerdo a la plantilla que se escogió anteriormente en la pantalla de configuración.

Figura DDDD : Plantilla de integración

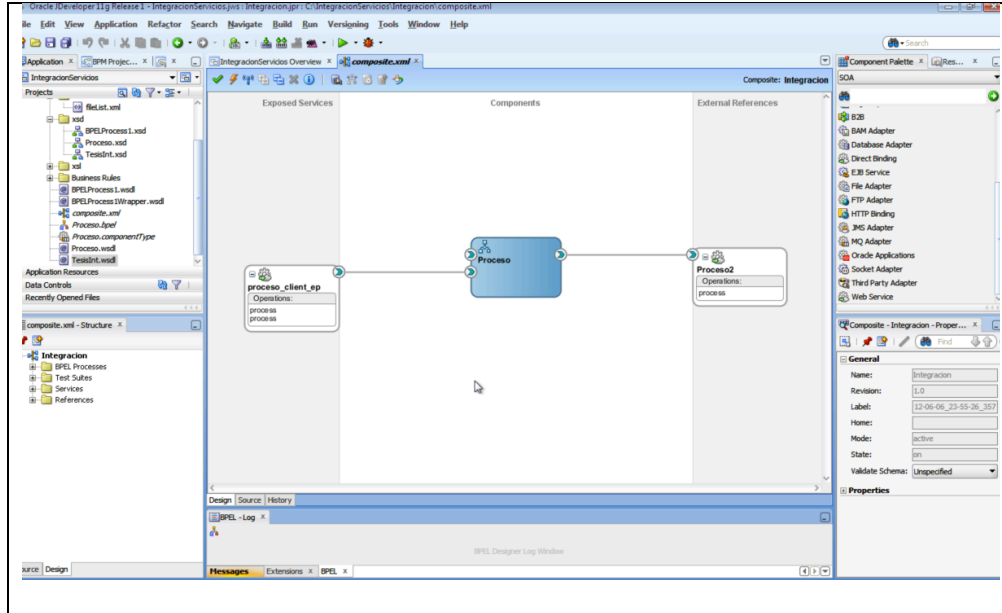


Fuente: Oracle Fusion Middleware SOA Suite

- Se añade el componente Service Web a la columna de referencias, para así tener la estructura de integración la cual consiste en un mensaje que

recorrerá el bus de servicio empresarial el cual irá a un servicio web para ser consumido.

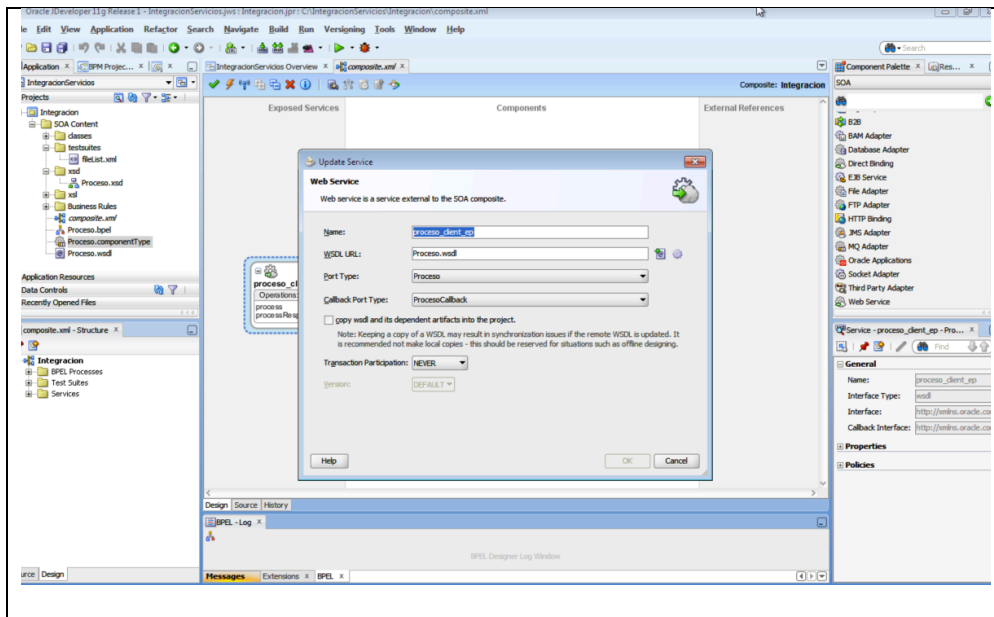
Figura EEEE : Estructura de servicios Web



Fuente: Oracle Fusion Middleware SOA Suite

- Al dar doble click sobre el servicio web se podrá definirlo ya sea de acuerdo a la dirección de la cual se va a consumir o del archivo propio del servicio web.

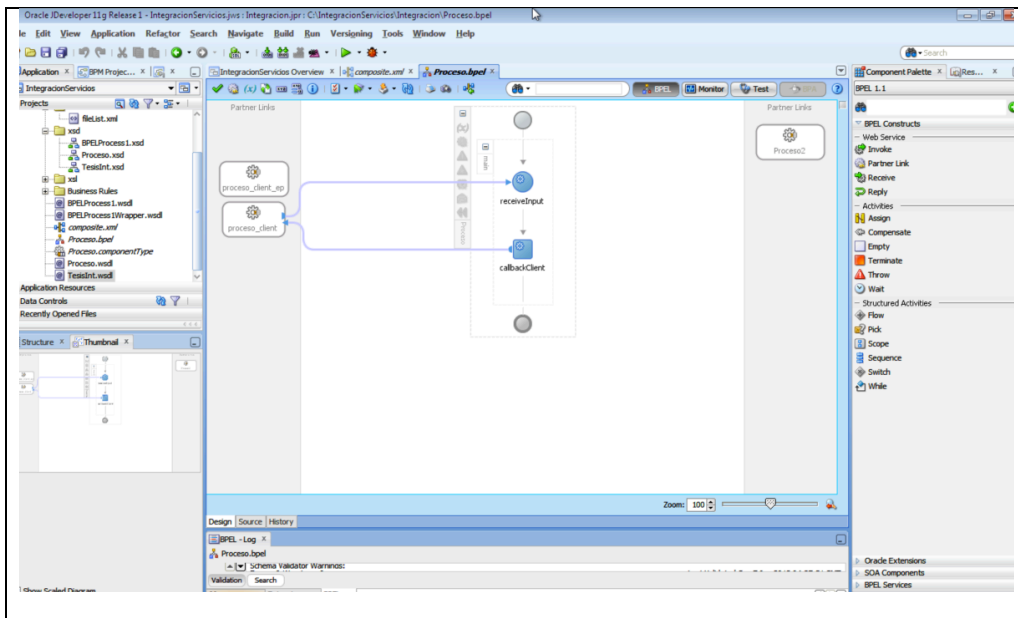
Figura FFFF : Propiedades del servicio web



Fuente: Oracle Fusion Middleware SOA Suite

- Al dar doble click en el conector de los servicios web se puede observar de manera detallada la estructura interna del conector y se realizará el mapeo entre servicios.

Figura GGGG : Propiedades del servicio web

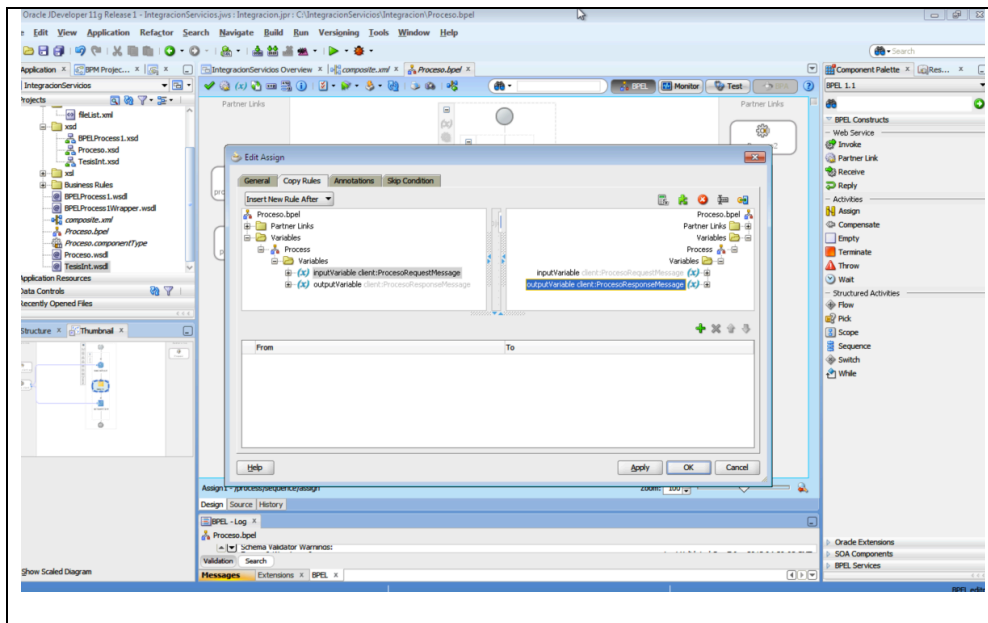


Fuente: Oracle Fusion Middleware SOA Suite

- Al ingresar en las propiedades tanto del servicio consumidor como del servicio a ser consumido se debe realizar el mapeo sobre cual servicio es

el que presentará los datos de salida y los datos de entrada, para así mediante funciones propias de JDeveloper realizar la operaciones necesarias de conversión de tipos de dato para que no se den errores en el intercambio de información.

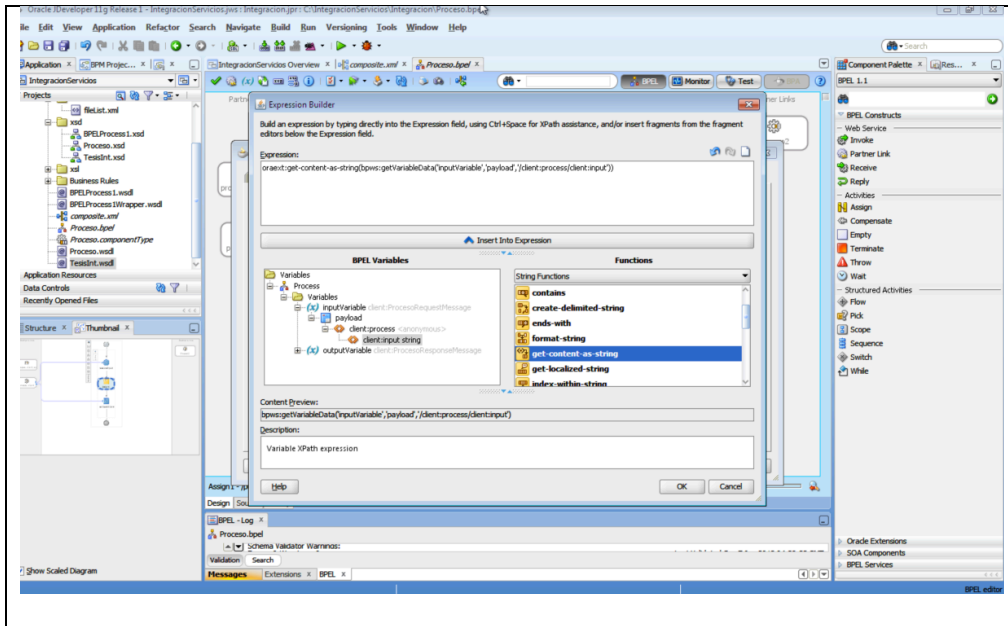
Figura HHHH : Propiedades de los servicios web a realizar el mapeo de datos



Fuente: Oracle Fusion Middleware SOA Suite

- En la columna Functions se escoge la función que se va a utilizar para la extracción de datos y para direccionar la información hacia el servicio web receptor del dato.

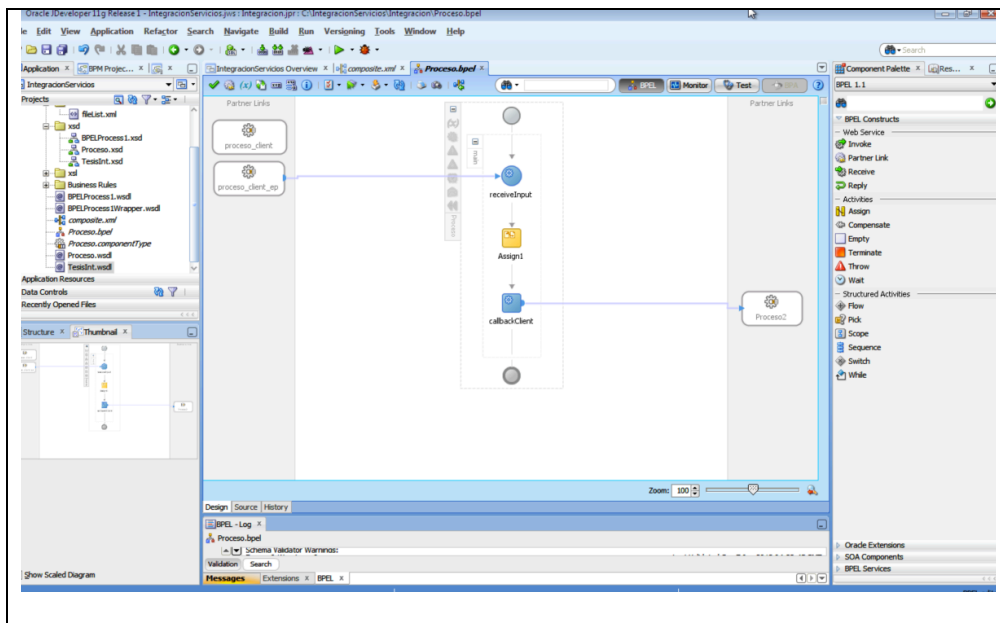
Figura III : selección de las funciones que actuarán sobre los datos



Fuente: Oracle Fusion Middleware SOA Suite

- Finalmente se podrá ver el resultado de la estructura de integración en el cual actuará de manera sincronizada sobre el bus de datos una orquestación de servicios web y que al tener un evento direccionará los mensajes hacia el destinatario.

Figura JJJJ : Estructura final de integración



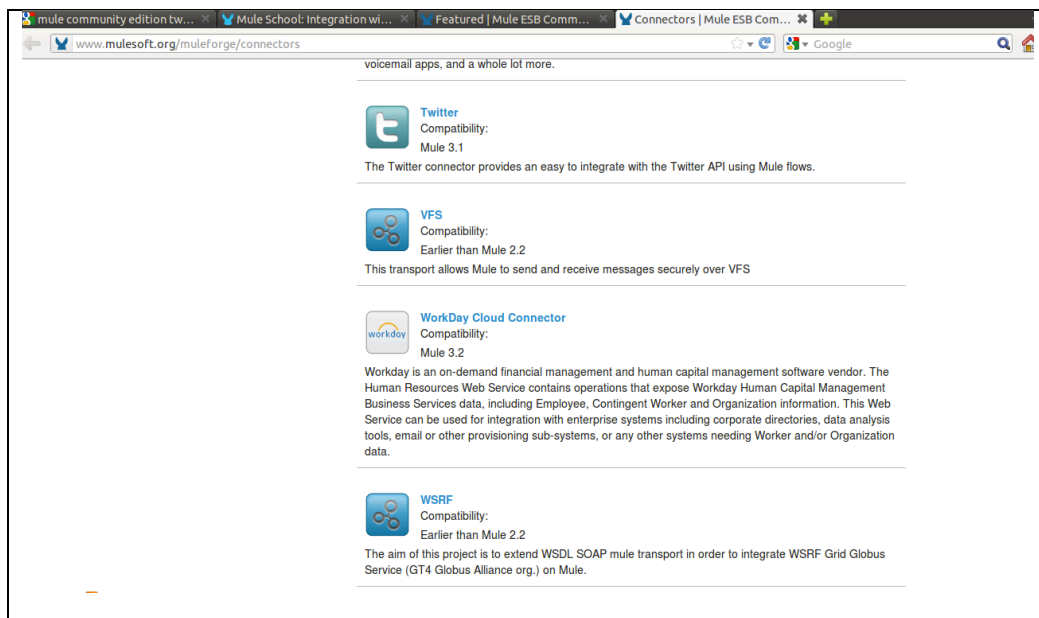
Fuente: Oracle Fusion Middleware SOA Suite

3.5.3. Integración de aplicaciones utilizando Mule® ESB Community Edition.

El escenario de pruebas para Mule ESB Community Edition logrará la integración de un servicio web que recibe como parámetro una variable tipo String y que se integrará mediante el Cloud Connector de la aplicación con la red social Twitter para la actualización del estado del usuario. Con esto se tendrá independencia de plataformas y se utilizará los servicios expuestos para ser consumidos por Twitter Developers⁵⁶.

- Se empezará descargando los conectores definidos MuleSoft para aplicaciones que se encuentran en la Nube (Cloud Computing). En la página <http://www.mulesoft.org/muleforge/connectors> se encuentran una serie de conectores para bases de datos, aplicaciones o servicios. El conector puede ser desarrollado por el usuario para aplicaciones que hayan sido generadas por la infraestructura de IT de la empresa ya que son bibliotecas creadas en Java.

Figura KKKK : Conectores para Mule ESB Community Edition

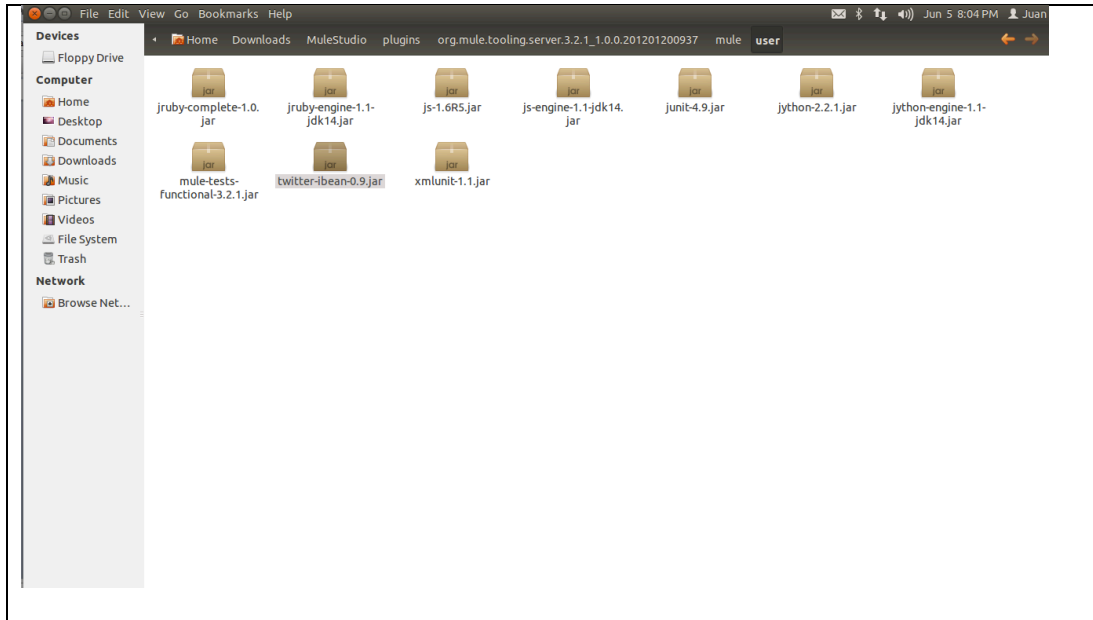


Fuente: Mule ESB Community Edition

⁵⁶ Twitter Developers es el grupo de desarrolladores de la red social Twitter en cuya página <https://dev.twitter.com/> se tiene acceso a los servicios expuestos de esta red social.

- El conector se debe ubicar en la carpeta de instalación de Mule ESB Community Edition en la carpeta Plugins.

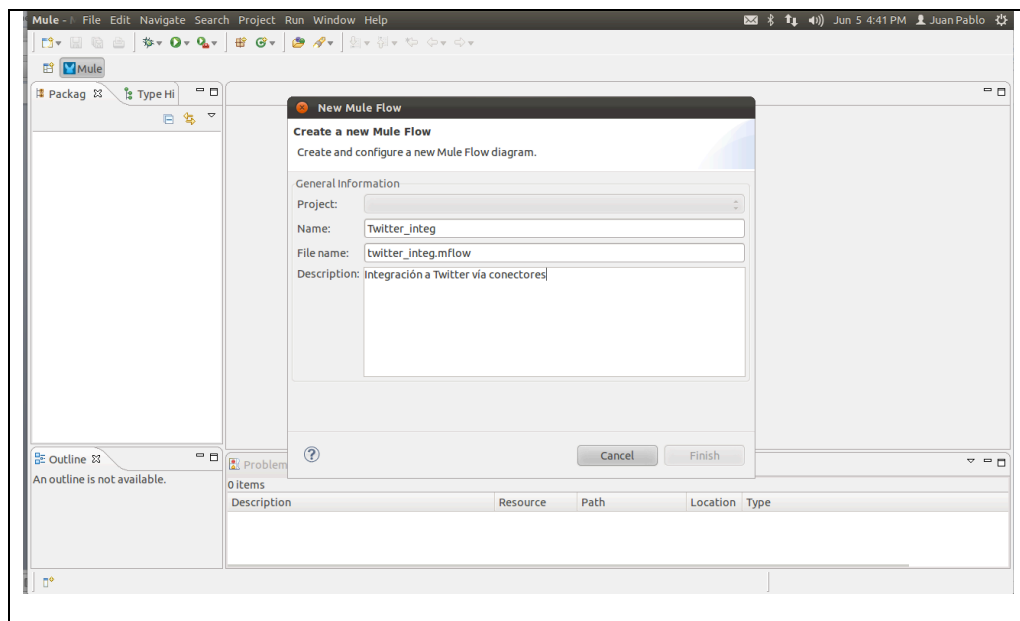
Figura LLLL : Ubicación del conector para Twitter



Fuente: Mule ESB Community Edition

- El siguiente paso es crear el flujo de integración en Mule ESB Community Edition en el cual se unirán los componentes y conectores. Para ello se debe ingresar en File, y escoger el sub menú New Mule Flow.

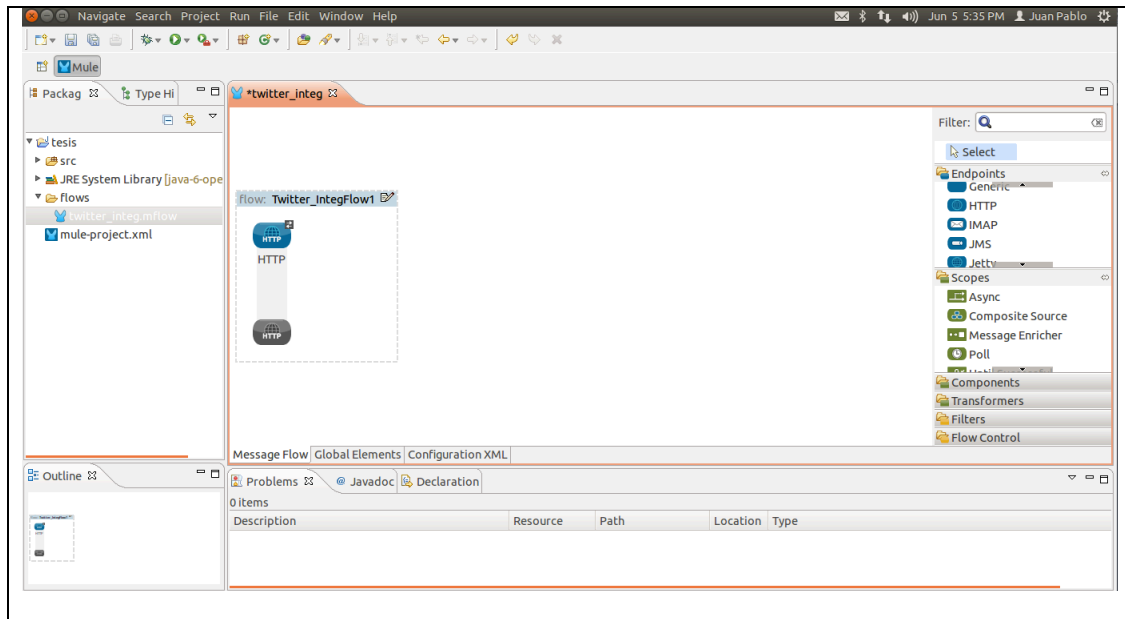
Figura MMMM : Creación de un flujo en Mule ESB Community Edition



Fuente: Mule ESB Community Edition

- Una vez creado el flujo se debe agregar los componentes y conectores que se encuentran en el menú derecho, con lo cual solo hay que arrastrar los componentes que se utilizarán en la integración. El primer componente registrará el método de una página http que al producirse un evento iniciará el flujo definido.

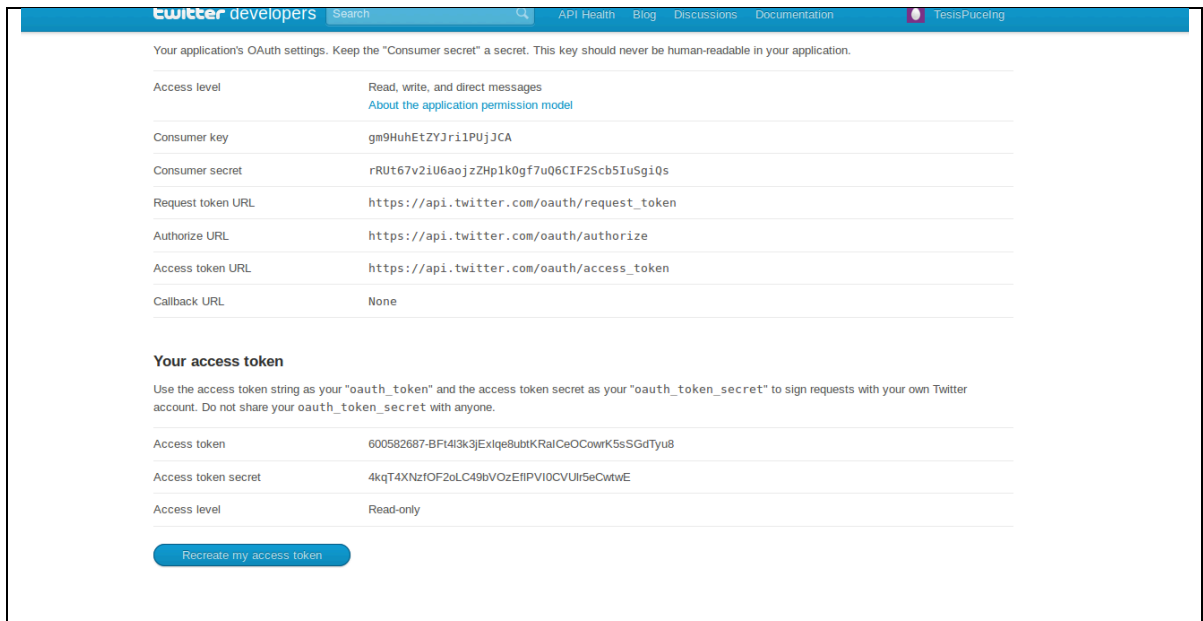
Figura NNNN : Componente HTTP de Mule ESB Community Edition



Fuente: Mule ESB Community Edition

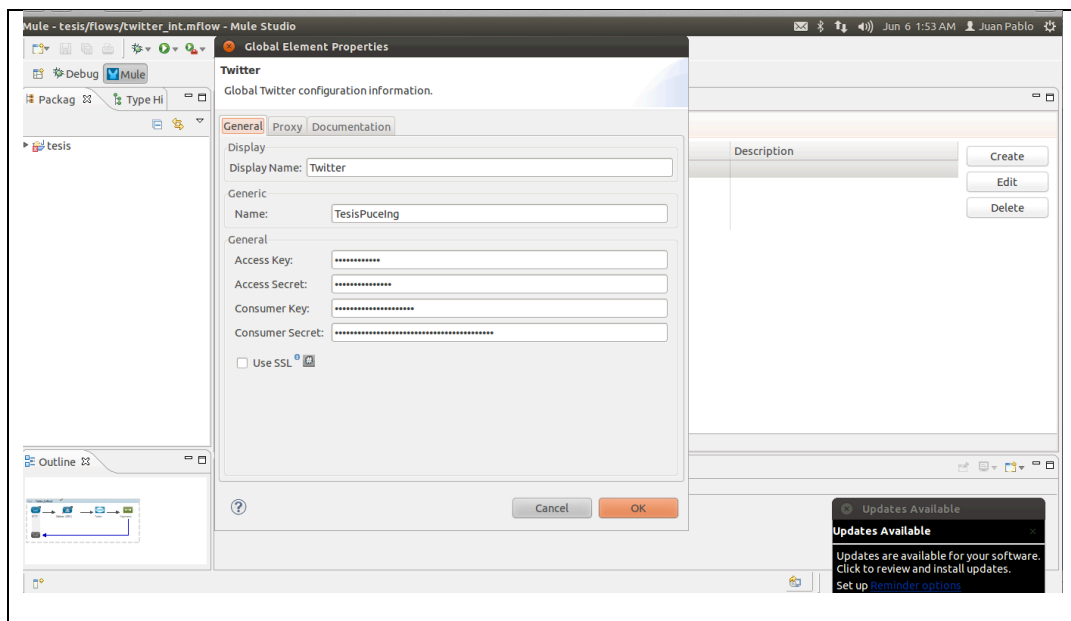
- Para añadir el conector que permitirá utilizar los servicios expuestos de Twitter se debe configurarlo en la pestaña Global Elements. En esta pantalla se debe ingresar datos fijos (ConsumerKey, ConsumerSecret) que permitirán manejar la cuenta a la cual se tendrá acceso. Estos datos se debe solicitar a Twitter en la página de desarrollo.

Figura OOOO : Página Twitter developers



Fuente: Mule ESB Community Edition

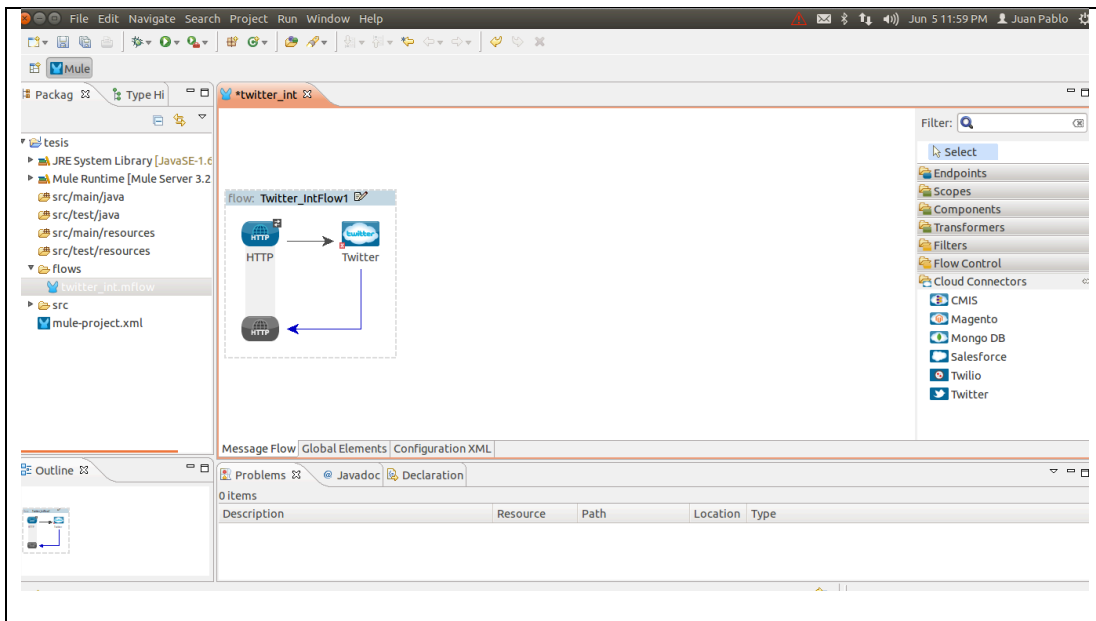
Figura PPPP : Parámetros del conector de Twitter



Fuente: Mule ESB Community Edition

- Ahora si se añadirá el conector de Twitter al flujo arrastrándolo desde el menú derecho de conectores hacia el espacio de trabajo

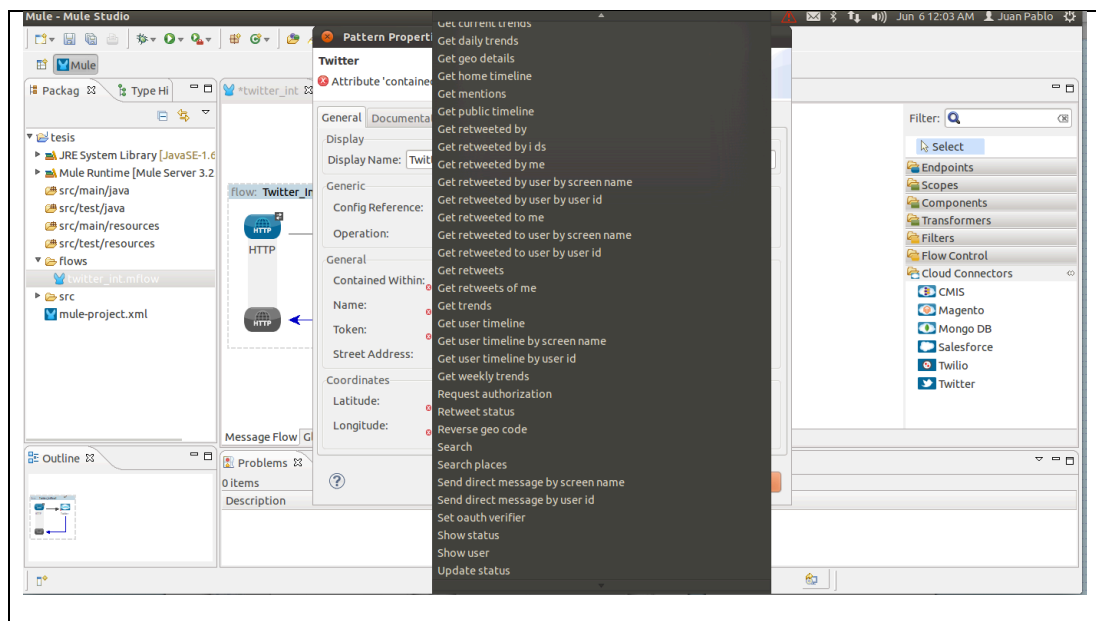
Figura QQQQ : Conector Twitter en el flujo



Fuente: Mule ESB Community Edition

- Al dar doble click sobre el conector de Twitter se debe escoger el método que se va a usar del servicio de twitter que se encuentra expuesto para ser consumido, en este caso se utiliza el método UpdateStatus.

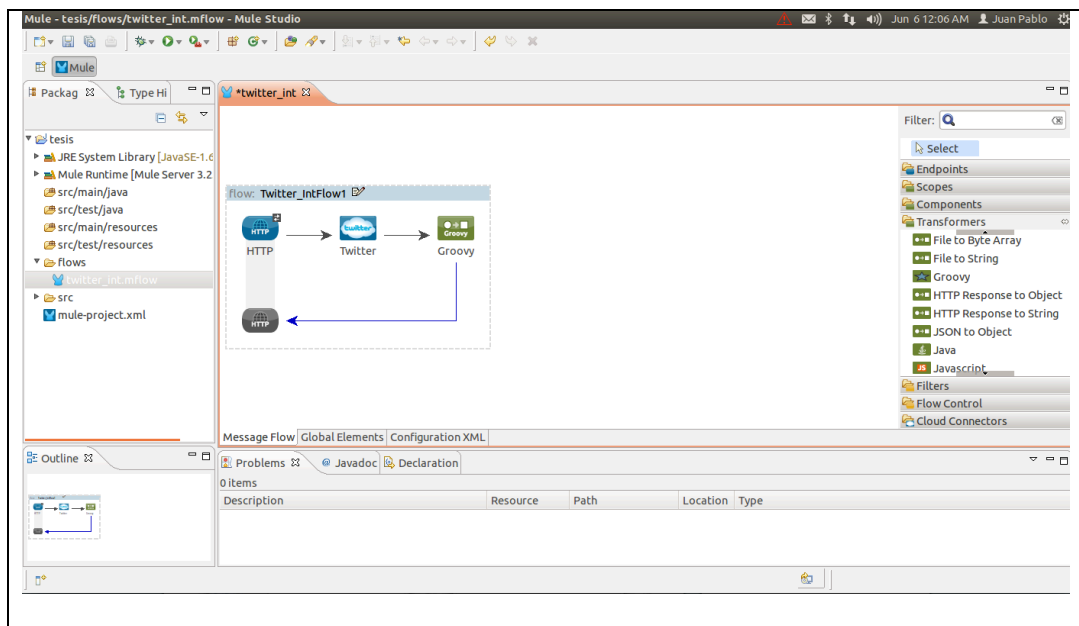
Figura RRRR : Métodos disponibles del conector Twitter



Fuente: Mule ESB Community Edition

- Se añadirá un componente de interpretación de mensajes de error que pueden presentarse durante la ejecución del flujo, en este paquete llamado Groovy se escogerá la función `payload.ToString()` que convertirá a formato string el error que devuelva el servicio Twitter.

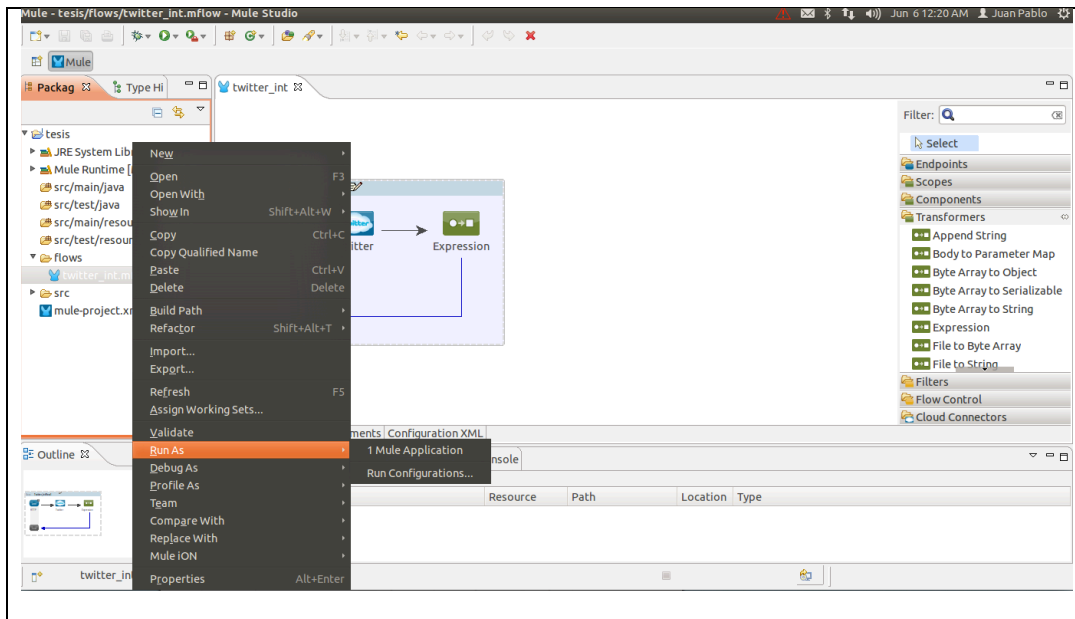
Figura SSSS : Interprete de errores en el servicio



Fuente: Mule ESB Community Edition

- Finalmente se escoge la opción del sub menú `Run As Mule Application` con lo cual se tendrá el flujo disponible cada vez que se presente el evento en el bus de servicio empresarial.

Figura TTTT : Ejecución del Flujo de Mule ESB para su disponibilidad



Fuente: Mule ESB Community Edition

3.6. Análisis del mantenimiento futuro de las herramientas de interfaces entre aplicaciones.

El mantenimiento que se deberá hacer a los componentes SOA de una integración de aplicaciones depende de la expansión de usuarios que se deseen o que se encuentren ocupando el software. El mantenimiento a las bases de datos que intervienen en la arquitectura de software SOA de la organización será el mantenimiento normal que se realiza sobre cualquier base de datos:

- Existen parámetros que permiten medir el rendimiento de una base de datos tales como el volumen máximo de transacciones que recibe la empresa, el tamaño usado por los tablespaces de la base de datos (especificar cuotas máximas de consumo de disco) y el backup de datos. Por lo que el mantenimiento se deberá hacer de acuerdo a los parámetros mencionados como por ejemplo eliminación de tablespaces innecesarios,
- La creación de índices sobre las bases de datos para facilitar el acceso a los datos almacenados y reducir los tiempos de búsqueda.

Las herramientas probadas en este proyecto de disertación poseen formas de mantenimiento similares entre las herramientas bajo el licenciamiento propietario (Microsoft Biztalk Server y Oracle Fusion Middleware SOA Suite)

| | Oracle Fusion Middleware SOA Suite | Microsoft Biztalk Server | Mule ESB Community Edition |
|--------------------------------|--|--|---|
| Mantenimiento necesario | <ul style="list-style-type: none"> Oracle Fusion Middleware posee actualizaciones de los conectores utilizados para unir las diferentes aplicaciones. Con esta actualización hay mejoras en el versionamiento de la aplicación y en el rendimiento. Node Manager posee la opción de administración y estadísticas en el cual se diagnostica el estado del bus de servicio empresarial para realizar el mantenimiento eliminando logs de la aplicación que no son utilizados. | <ul style="list-style-type: none"> El decrecimiento de rendimiento programado para esta herramienta es a partir de los 5GB de tamaño de las bases de datos, en ese momento se debe iniciar con el mantenimiento que el paquete ofrecido por Microsoft facilita. El agente que permite realizar el mantenimiento a la estructura del bus de servicio empresarial se llama BiztalkDTADB el cual realizará una limpieza de los datos propios de Biztalk server como por ejemplo el historial de mensajes de la aplicación. La actualización de conectores que Microsoft ofrece para | <ul style="list-style-type: none"> El mantenimiento de la estructura SOA en Mule ESB Community Edition se lo debe realizar sobre las bases de datos que intervienen en la solución puesto que este software no posee opciones de mantenimiento, tan solo la actualización de conectores. |

| | | | |
|--|--|--|--|
| | | otras bases de datos también forma parte del mantenimiento de la solución. | |
|--|--|--|--|

Tabla 24 : Mantenimiento de las herramientas

CAPÍTULO IV

4. Caso de análisis: Integración de bases de datos financiera y de operaciones de la campaña de perforación en Petroamazonas EP.

4.1. Descripción de la situación real.

Petroamazonas EP mantiene entre sus activos sistemas administrativos que permiten controlar la parte operativa y financiera de la campaña de perforación en cada taladro que la empresa posee. Para la parte operativa Petroamazonas EP maneja el software Openwells y para la parte financiera maneja Oracle Financials, cada uno de estos sistemas es independiente por lo que se presentaron discrepancias de datos entre ellos.

En el sistema Oracle Financials se almacenan los términos contractuales que las empresas de servicios petroleros han acordado con Petroamazonas EP y generalmente es manejado por el departamento financiero en Quito mientras que Openwells es manejado por el Ingeniero Junior de Perforación en donde se encuentre realizando el trabajo de construcción del pozo y se almacenan los datos que facturan las empresas de servicio a la par que han realizado el trabajo.

Debido a la independencia entre sistemas las tarifas que estaban acordadas en Oracle Financials se cumplían rara vez, en cambio; en Openwells se almacenaban los costos reales de construcción de cada pozo. Estas discrepancias hacían imposible realizar un estimado real del costo de un pozo de acuerdo a los contratos firmados y muchos servicios no podían ser monitoreados.

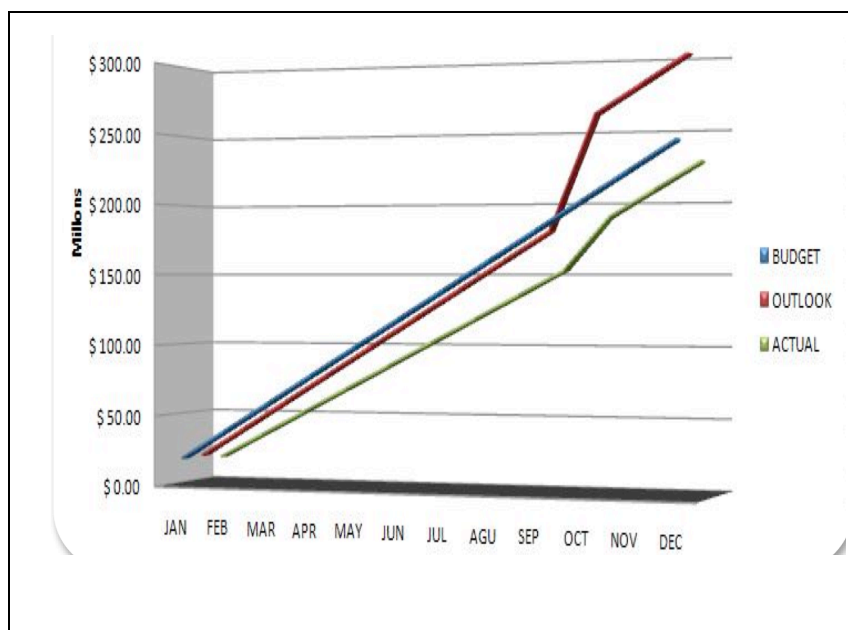


Figura UUUU : Presupuesto vs. situación real

4.2. Características de las aplicaciones.

Las aplicaciones a integrar tienen el licenciamiento propietario, Oracle Financials permite el manejo de las tablas de la base de datos y OpenWells de la misma manera, no son sistemas cerrados.

OpenWells. Es un sistema desarrollado por Halliburton – Landmark integrado y comprensivo de las operaciones de pozo, administración de datos corporativos para el análisis de información vital para ingeniería. OpenWells soporta los reportes de perforación y reacondicionamiento dentro de la misma aplicación para el pozo en operación.

Oracle Financials. Es un sistema financiero para el manejo y control de cuentas, manejo de contratos, creación de reglas de contratos el cual es usado por Petroamazonas EP para realizar los pagos a las empresas que prestan servicios para la campaña de construcción de pozos.

4.3. Requerimientos de integración.

Los costos ingresados en la base de datos de operaciones de Openwells deberán encontrarse estrictamente bajo las reglas de los contratos almacenados en Oracle Financials, y la información de servicios recibidos que se ingresa en Openwells

sea a su vez ingresada en Oracle Financials para la correcta emisión del pago hacia las empresas que prestan servicios en los diferentes taladros en las que se encuentran.

4.4. Solución.

- Debido a que en Petroamazonas EP se manejaban las órdenes de trabajo sobre un pozo en papel para luego ser reflejadas en el sistema Oracle Financials que es el encargado de realizar la factura para que las empresas contratistas cobren por su trabajo, se decidió realizar el sistema CORE Management ⁵⁷ para ello se estructuraron los flujos de trabajo y los procesos que involucran la campaña de perforación de un pozo.
- CORE Management es un sistema de interface entre Oracle Financials y Openwells, es un sistema de integración para así evitar las inconsistencias en los datos que se presentaban en las fechas anteriores a su desarrollo.
- CORE Management toma los términos contractuales de las empresas de servicio almacenados en Oracle Financials para validar la información a ingresar en Openwells.
- El costo diario de perforación de un pozo y el detalle de los servicios involucrados en la perforación se almacenan en la tabla DM_DAILYCOST del sistema Openwells en la cual se aceptarán los datos que han sido validados con los contratos almacenados en Oracle Financials en la tabla PO_REQ_DISTRIBUTIONS_ALL.
- El paso de información mediante el ESB (Bus de Servicio Empresarial) y que es donde interviene el software de integración se lo realiza únicamente en las fechas de finalización del pozo (una vez por mes y para cada pozo) en el cual la información de la tabla DM_DAILYCOST de Openwells irá a la tabla de Oracle Financials PO_REQUISITIONS_INTERFACE_ALL con la cual es generada la factura y pagos para las empresas contratistas que intervinieron en los pozos.

⁵⁷ CORE Management. Sistema de Reducción de costos desarrollador por Halliburton Landmark y Petroamazonas EP.

- Luego de los primeros 3 meses de operación de CORE Management se obtuvo igualdad y consistencia de los datos de los sistemas Openwells y Oracle Financials y una reducción de costos de 1.5 millones de dólares en cada torre de perforación⁵⁸.

⁵⁸ http://www.halliburton.com/public/landmark/contents/Case_Histories/web/H09286_Spanish.pdf

CAPÍTULO V

5. Conclusiones y recomendaciones.

5.1. Conclusiones.

1. Trabajar con una arquitectura orientada a servicios requiere un alto nivel de abstracción, para decidir que procesos se van a exponer como servicios dentro del flujo del negocio, para ello se debe conocer la lógica del negocio y así observar los procesos que van a ser utilizados de manera continua dentro de la empresa así como los procesos que se necesitan exponer para ser utilizados por los clientes y filiales de la empresa.
2. Implementar un bus de servicio empresarial dentro de la organización implica que se hayan definido de manera lógica los procesos de la empresa y la correcta arquitectura orientada a servicios para así ver la metodología de transmisión de datos y de ser necesario la búsqueda o creación de conectores para acceder a las aplicaciones expuestas en el bus de servicio empresarial.
3. La integración de sistemas se debe realizar entre aplicaciones que intervengan en el flujo de trabajo del negocio y cuando haya riesgo de caer en los problemas mencionados en la investigación como lo son la duplicación de datos o las diferencias en los resultados, ya que una integración entre aplicaciones que no son importantes para los procesos del negocio, implica la obtención de resultados que no son imprescindibles para la toma de decisiones de la organización.
4. Mule ESB Community Edition posee gran soporte de parte de la comunidad de usuarios, encontrándose en la red gran cantidad de ejemplos prácticos lo cual hace que esta herramienta libre sea muy competitiva con respecto al soporte de las herramientas propietarias utilizadas en esta investigación.
5. Si se desea tener una aplicación de integración con flexibilidad en el servidor de aplicaciones, no se debería optar por Oracle Fusion Middleware SOA Suite debido al sustancial incremento de costos del licenciamiento al no usar un servidor de aplicaciones Oracle.
6. Oracle Fusión Middleware SOA Suite permite la instalación sobre una amplia gama de sistemas operativos mientras que Microsoft Biztalk Server

solo podrá ser instalado en sistemas operativos de la familia Microsoft Windows. Mule ESB Community Edition permite la instalación bajo sistemas operativos Windows, Linux y OSX.

7. Tomando en cuenta el aspecto económico de la empresa, si es una empresa con presupuesto limitado Mule ESB Community Edition es la mejor opción para integración ya que el licenciamiento no tiene costo y no hay restricciones en el crecimiento de las aplicaciones a integrar por lo que el cambio de aplicación en el futuro será opcional.
8. El mantenimiento de las aplicaciones depende de las bases de datos que intervienen en la integración, por lo que las 3 herramientas analizadas poseen similares regímenes de mantenimiento y de acuerdo a este aspecto se podrá utilizar cualquiera de ellas.

5.2. Recomendaciones.

1. Al utilizar una herramienta bajo el licenciamiento libre se debe involucrar con la comunidad que la desarrolla ya que así se obtendrá soporte valioso en los foros que crean las personas involucradas tanto en el desarrollo como en la utilización de software con este tipo de licenciamiento.
2. Es preferible invertir recursos en realizar una investigación para ver si nuestra empresa necesita una integración de aplicaciones, o si es el momento adecuado de realizar la integración de sistemas en lugar de empezar con la integración y no obtener una mejora en el flujo de trabajo de la empresa.
3. Si se desarrolla sistemas, se debe enfocar en realizarlo bajo la arquitectura orientada a servicios ya que así se tiene un entorno de software desacoplado y por lo tanto de fácil integración con otros sistemas y, además; nos permitirá compartir los servicios de la empresa con clientes o con filiales.
4. Para exponer un servicio web a los clientes y/o filiales de la empresa se debe definir las funciones que este servicio va a cumplir, ya que no se debe exponer todos los servicios debido a que muchos sirven únicamente para la lógica interna del negocio y presentarlos para ser consumidos

externamente implicaría problemas de seguridad, gasto de recursos sin obtener una ganancia en funcionalidad.

5. Al realizar la descarga de los archivos necesarios para la instalación de Oracle Fusion Middleware SOA Suite, debido a su tamaño, debemos utilizar un programa que chequee la integridad de los archivos comprimidos descargados ya que es muy común que estos tengan errores, y en el momento de la instalación no nos permita continuar con normalidad la misma.
6. Se debe revisar casos exitosos de implementaciones antes de tomar una decisión acerca del modo de licenciamiento a utilizar, ya que muchas veces nuestros requerimientos de integración pueden ser solucionados con software bajo el licenciamiento libre.

6. Bibliografía.

- Almohalla, Enrique. “Estrategias y metodologías postimplantación en la gestión de ERP’s”. Instituto de empresa. (Marzo de 2006).
- Articura Education Inc. SOA Standards and Specifications. Internet. <http://www.whatissoa.com/soaspecs/default.php>. 20 de Enero de 2012.
- Atlassian Confluence Open Source Project. Java Business Integration. Internet. <http://doc.petalslink.com/display/petalsesb30/Java+Business+Integration>. Acceso: 5 de abril de 2012.
- Cejas, Julio. Mule Enterprise Service Bus. Internet. <http://www.slideshare.net/jcejas/mule-esb-introduccion>. Acceso: 2 de Mayo de 2012.
- Dapozo, Gladys. Tecnologías Emergentes Multiplataforma. Internet. <http://www.sicuma.uma.es/sicuma/independientes/argentina08/Dapozo-Litwak/index.htm>. Acceso: 20 de Abril de 2012.
- Díaz, Juan Pablo. “SOA Pragmático: Visión de Sun Microsystems”. ACIS. (27 de Septiembre de 2005) Bogotá, 2005.
- Kok, Yuri. Open Source SOA. Internet. <http://www.slideshare.net/industrialtsi/sopera-opensource-soa>. Acceso: 20 de Abril de 2012.
- Martínez, Marcela. Modelo de madurez en SOA. Internet. http://www.scribd.com/ealtamirano_1/d/63956169-1-Modelo-de-Madurez-Soa. Acceso: 8 de febrero de 2012.
- Microsoft Corporation. “La arquitectura orientada a servicios de Microsoft aplicada al mundo real”. (14 de Mayo de 2007) Seattle, 2007.
- Murillo, Mauricio. Integrando aplicaciones con Mule. Internet. <http://www.slideshare.net/ingmmurillo/integrando-aplicaciones-con-mule-barcamp2010>. Acceso: 5 de Mayo de 2012.
- Pérez, Ramón. “Servicios Web, Orquestación y Coreografías”. Universidad de Oviedo.(Marzo de 2008) Oviedo, 2008.
- Petritsch, Helmut. “Service oriented architecture vs. Component based architecture”. (2010) London.

- Roden, Peter. "The use of standards in SOA". The Second Service Oriented Architecture and Web Services Best Practices. (18 de octubre de 2005) Boston, 2005.
- Rodríguez, Wladimir. "Integración de aplicaciones". Posgrado en computación ULA.(2009) Caracas, 2009.
- Schwed, Alejandro. "El modelo de integración basado en servicios SOA". Bolsa de valores de Colombia. (26 de septiembre de 2007) Bogotá, 2007.
- Woolf, Bobby. Enterprise Integration Patterns. Internet. <http://www.enterpriseintegrationpatterns.com/toc.html>. Acceso: 12 de Marzo de 2012.

7. Glosario de términos técnicos y siglas.

API. Application Programming Interface, por sus siglas en inglés, es un código fuente diseñado para permitir la comunicación de una aplicación a otra.

Backbone. Es una infraestructura diseñada para el transporte de datos en una red

BPEL. Bussines Process Execution Language, por sus siglas en inglés, es un lenguaje ejecutable para especificar acciones entre procesos de negocio y servicios web.

BPEL. Bussines Process Execution Language, por sus siglas en inglés, es un lenguaje ejecutable para especificar acciones entre procesos de negocio y servicios web.

BPM. Business Process Management, por sus siglas en inglés, se lo considera una evolución de los sistemas de workflow y el tratamiento automatizado de los procesos de negocio de las empresas.

BPMN. Bussines Process Model and Notation, por sus siglas en inglés, es una representación gráfica de procesos de negocio en un modelo de procesos de negocio.

CORBA. Common Object Request Broker Architecture.

CRM. Sistemas de administración de clientes por sus siglas en inglés (Customer Relationship Manager)

DCOM. Distributed Component Object Model.

EJB. Enterprise Java Bean.

ERP. Sistemas de planificación de recursos empresariales por sus siglas en inglés (Enterprise Resource Planning)

ESB. Modelo para diseñar e implementar la interacción entre aplicaciones de software en la arquitectura orientada a servicios.

HR. Sistemas de recursos humanos por sus siglas en inglés (Human Resources)

IIOB. Internet Inter-ORB protocol, Protocolo que funciona bajo TCP/IP

Middleware. Interface de unión entre componentes de software

MOM. Message Oriented Middleware.

POJO. Plain Old Java Objects.

SOA. Service Oriented Architecture.

SOAP. Simple Object Access Protocol.

UDDI. Universal Description Discovery and Integration.

Wrapper. Grupo de subrutinas o clases utilizadas para desarrollar software y que se encuentran empaquetadas mostrando únicamente su funcionalidad.

WSDL. Web Services Description Lenguaje.

XML. Extensible Markup Lenguaje.

XPDL. Process Definition Language por sus siglas en inglés, fue diseñado para intercambiar tanto la parte gráfica como semántica de un flujo de proceso del negocio.

YAWL. Yet Another Workflow Language, por sus siglas en inglés, es un lenguaje de flujo de datos aceptado en software bajo licencia Open Source.