Integración de fuentes de datos, tecnologías semánticas y bus de servicios empresarial

***Johnny C. Solórzano Z.*[1], *Víctor Saquicela*[2]**

[1] Maestría en Gestión Estratégica de Tecnologías de la Información, Universidad de Cuenca, Av. 12 de Abril y Av. Loja, Cuenca, Ecuador, 01.01.168

[2] Departamento de Ciencias de la Computación, Universidad de Cuenca, Av. 12 de Abril y Av. Loja, Cuenca, Ecuador, 01.01.168

Autor para correspondencia: victor.saquicela@ucuenca.edu.ec

Fecha de recepción: 30 de julio de 2017 - Fecha de aceptación: 15 de agosto de 2017

RESUMEN

Actualmente las organizaciones disponen de diferentes sistemas de información para soportar su modelo de negocio. Generalmente la incorporación de estos sistemas se ha realizado sin seguir un proceso riguroso para determinar la interoperabilidad con los sistemas pre existentes. Por otra parte, la aparición de nuevas tecnologías, han llevado a adoptar diferentes plataformas de software, lo que ha ocasionado que se tengan islas de información dentro de una misma organización. En este entorno, las organizaciones se encuentran limitadas para la explotación de datos, la integración de procesos, así como en la producción de conocimiento e información para toma de decisiones. Es común encontrar sistemas informáticos que operan de manera autónoma, esto implica que muchos procesos se encuentren desacoplados y la extracción de información se convierte en un verdadero reto puesto que el traspaso de datos entre aplicaciones se realiza utilizando scripts, archivos de texto y, en algunos casos, de forma manual, lo cual implica pérdida de tiempo, riesgo de datos inconsistentes y por supuesto sobrecarga de horas de trabajo tanto para personal técnico como administrativo. Basado en la problemática descrita, la principal contribución de este trabajo consiste en proponer un modelo de integración utilizando Tecnologías Semánticas y un Bus de Servicios Empresarial, que permita tener una visión unificada de las fuentes de datos en todas las operaciones transaccionales ejecutadas en una organización.

Palabras clave: ESB, ontologías, integración de datos, integración semántica, CC.

ABSTRACT

Traditionally, organizations have different information systems to support their business model. Generally, the incorporation of these systems has been carried out without following a rigorous process to determine the interoperability with the preexisting systems. On the other hand, the appearance of new technologies led to the existence of different software platforms each with their own specific information. In this environment, organizations are limited to data mining, process integration, as well as the production of knowledge and information for decision making. It is common to find computer systems that operate autonomously, this implies that many processes are decoupled, and the extraction of information becomes a real challenge since the transfer of data between applications is done using scripts, text files and in some cases manually which implies loss of time, risk of inconsistent data, and of course overload of working hours for both technical and administrative staff. Based on the described problem, the main contribution of this work is to propose an integration model using Semantic Technologies and Enterprise Service Bus, which allows a unified view of data sources in all transactional operations executed in an organization.

Keywords: ESB, ontologies, data integration, semantic integration, CC.

1. INTRODUCCIÓN

De acuerdo con Beyer, Thoo, Zaidi, & Greenwald (2016), la integración de fuentes de datos cobra importancia precisamente por la existencia de un gran volumen y variedad de datos. Por lo tanto, poder gestionar adecuadamente esa información e interpretarla correctamente otorga una ventaja competitiva más que necesaria en la actualidad. La integración de fuentes de datos pretende definir arquitecturas, modelos e infraestructura de software para permitir a los usuarios acceder a datos almacenados en fuentes de datos heterogéneas, presentando una vista unificada de los sistemas de información. Uno de los grandes problemas a la hora de transformar la información disponible en conocimiento, es la diversidad de estructuras y formatos de la información de partida. Para alcanzar la mayor eficacia, es necesario integrarla bajo una estructura común homogénea, como paso previo a su empleo.

Conseguir la colaboración entre sí de aplicaciones distribuidas, heterogéneas y posiblemente autónomas, es un gran reto en el área de la integración de fuentes de datos. En este contexto, distribución, hace referencia a las diferentes aplicaciones que pueden ejecutarse en máquinas conectadas a través de una red (LAN o Internet). Autonomía se refiere a que el sistema de integración no puede esperar que la aplicación cambie su forma de actuar para facilitar la integración (aplicaciones heredadas, aplicaciones de otros departamentos, aplicaciones de otras empresas, etc.). Finalmente, heterogeneidad, hace referencia a los diferentes tipos de software o hardware utilizados y sobre los cuales se ejecutan las aplicaciones.

El presente trabajo tiene por objetivo i) caracterizar los problemas de la integración de la información en general, ii) identificar las propuestas descritas en la literatura sobre el proceso de integración de sistemas de información heterogéneos, y iii) proponer una solución al problema de integración de las organizaciones utilizando tecnologías semánticas. Para llevar a cabo la consecución de estos objetivos, se propone el despliegue de una solución prototipo de integración de fuentes de datos, que permita: 1) minimizar los requerimientos computacionales, 2) lograr una arquitectura simple, basada en estándares, 3) poder integrar fuentes de datos heterogéneas, y 4) ejecutar de manera eficiente consultas distribuidas que permitan realizar uniones entre datos provenientes de diferentes fuentes.

El documento está organizado de la siguiente forma: en la Sección 2, se describe brevemente el problema de la integración de datos, en donde se caracteriza los principales problemas, enfocándose en los diferentes niveles de heterogeneidad que deben ser considerados al momento de proponer una solución, por otra parte, se realiza un análisis teórico de las diferentes formas para abordar los problemas de integración, además se describen algunos trabajos relacionados. En la Sección 3, se describe un escenario sobre el que se aplicará la propuesta de integración de este trabajo. En la Sección 4, se discute las soluciones de integración propuestas. En la Sección 5, se muestra la propuesta recomendada. Finalmente, algunas conclusiones de este trabajo son discutidas en la Sección 6.

2. ANTECEDENTES Y TRABAJOS RELACIONADOS

2.1. Antecedentes

Lenzerini (2002) define la integración de datos como “El problema de combinar datos residentes en diferentes fuentes y proveer al usuario una vista unificada de esos datos”. Actualmente, tanto empresas medianas como grandes mantienen datos de manera heterogénea, procedentes de un amplio conjunto de fuentes (sistemas: financieros, logística, talento humano; hojas de cálculo, archivos de texto, etc.). De acuerdo con diferentes análisis presentados por Bernstein & Haas (2008) acerca de los sistemas de información empresariales, se puede desprender que estos no han sido diseñados para integrar datos o aplicaciones a posteriori. Alcanzar con éxito una vista homogénea sobre datos de diferentes fuentes en una organización dependerá de: 1) el contenido y funcionalidad de los actuales sistemas de información, 2) la clase de información que es manejada por los distintos sistemas, 3) la intención de

uso de los sistemas de información, y 4) la disponibilidad de recursos (tiempo, recursos humanos, dinero, etc.). Estos factores han sido considerados en este trabajo para proveer una solución que intente balancear el costo-beneficio de una plataforma que permita la integración de múltiples fuentes de datos en las organizaciones.

La integración de múltiples sistemas de información tiene en general como objetivo combinar los sistemas seleccionados de manera que forman un todo unificado y ofrezcan a los usuarios la sensación de interactuar con un solo sistema de información. Para ello, todos los datos tienen que ser representados con los mismos principios de abstracción (modelo unificado de datos y semántica). Esta tarea incluye la detección y resolución de conflictos de heterogeneidad a múltiples niveles como: estructuración, modelo de datos, plataforma de software, convenciones semánticas, convenciones sintácticas, diferencia de granularidad. Para satisfacer las necesidades de integración que serán descritas en esta sección se analiza diferentes propuestas. La clasificación usada para efectuar el análisis está basada en el trabajo introducido por Dittrich & Jonscher (1999) donde se distinguen propuestas de integración según el nivel de abstracción.

Los sistemas de información pueden describirse usando una arquitectura de capas (Tsierkezos, 2010), como se muestra en la Figura 1. En la capa superior, los usuarios acceden a los datos y servicios a través de diversas interfaces que se ejecutan en la parte superior de diferentes aplicaciones. Las aplicaciones pueden utilizar un middleware, monitores de procesamiento de transacciones (TP), middleware orientados a mensajes (MOM), SQL middleware, etc. para acceder a datos a través de una capa de acceso de datos. Los propios datos son gestionados por un sistema de almacenamiento de datos. Por lo general, son usados sistemas gestores de base de datos (DBMS) para combinar los datos de acceso y la capa de almacenamiento. En este contexto se han analizado las posibles soluciones para integración de fuentes de datos: 1) Integración Manual, 2) Interfaz de Usuario Común, 3) Integración de Aplicaciones, 4) Integración por Middleware, 5) Acceso Uniforme de Datos, y 6) Almacenamiento de Datos Común.

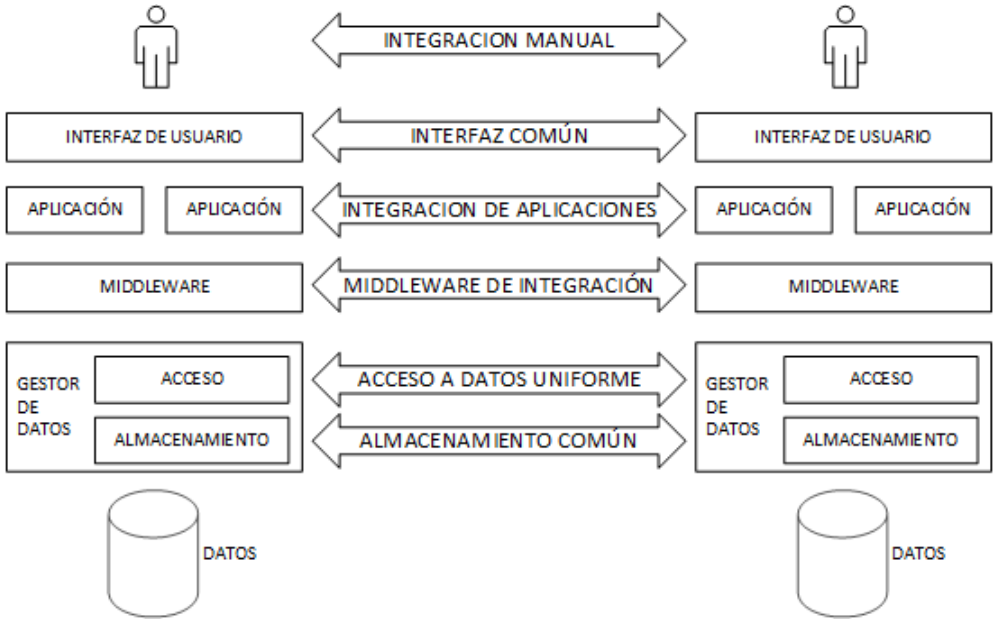


Figura 1. Tipos de integración.

En este trabajo se aplica un enfoque de acceso de datos uniforme utilizando un Bus de Servicios Empresarial (ESB) y Tecnologías Semánticas que provean un simple punto de acceso para acceder a las consultas de varias fuentes de datos. Un mediador que contenga un procesador global de consultas será empleado para enviar subconsultas a las fuentes de datos locales. Más detalles de esta propuesta serán descritos en las siguientes secciones.

Se descarta el uso de las otras propuestas de integración debido a que, en el caso concreto de las dos primeras opciones, estas representan un esfuerzo enorme de implementación, pero sobre todo no permite escalar la solución a futuros sistemas que se integren en la organización. La integración

usando middleware no fue considerada como una opción puesto que esta solución requeriría la incorporación de diferentes herramientas para construir un sistema integrado, y, debido a la heterogeneidad de los sistemas actuales en las organizaciones, representa un proyecto de muy alto costo y riesgo. Finalmente, la opción de integración mediante un repositorio común es una buena alternativa para los futuros sistemas de información que se integren a la organización, sin embargo, la gran diversidad de datos y esquemas encontrados en los sistemas actuales hacen inviable esta solución por su costo y tiempo de implementación.

2.2. Trabajos relacionados

Roa-Valverde & Aldana-Montes (2008) proponen el uso de un ESB combinado con tecnologías de Web Semántica con la finalidad de crear un sistema "inteligente" que facilite la integración de datos. La idea es agregar un módulo semántico a la funcionalidad del ESB de manera que se puedan anotar semánticamente los objetos que son manejados dentro del Bus. Esto proporciona a los artefactos implementados en el Bus la capacidad de razonar, facilitando tareas como el Descubrimiento y Composición de servicios.

Shi, Gao, Xu, & Xu (2014) presentan una plataforma de integración basada en la teoría de ontologías y un ESB. Su propuesta indica que la integración se puede lograr haciendo un mapeo entre los modelos ontológicos de datos, para lo cual utilizan la técnica de mapeo de 3 capas; y, que las nuevas demandas se pueden atender mediante la reorganización de servicios.

Jin, Lv, & Xiang (2014) proponen el uso de tecnologías semánticas para mejorar las capacidades de un ESB. Ellos manifiestan que los ESB actuales únicamente permiten describir los servicios que proporcionan de manera sintáctica, lo cual imposibilita una mediación de servicios semántica y el razonamiento sobre los datos a integrar. Por esta razón proponen la incorporación de anotaciones semánticas basadas en ontologías con la finalidad de enriquecer y conciliar la semántica de los datos que circulan a través del ESB.

Schratzenstaller, Baldo, & Rabelo (2016) plantean establecer un método de apoyo semántico para permitir que los datos generados por empresas PYMES se integren de forma flexible y automática utilizando servicios web a través de un ESB. Este enfoque supone que un BPM (Business Process Model) describe la orquestación de los autores participantes (PYMES) a través de servicios predefinidos y propone un método para la integración por BPM. Así se resuelven las incompatibilidades sintácticas y semánticas que existen entre el servicio definido en el BPM, y que son implementadas efectivamente por la PYME. Se supone como hipótesis que el soporte semántico puede añadirse creando una representación de los conceptos y metadatos proporcionados por las empresas participantes.

Harcuba & Vrba (2015) plantean una solución para integrar clientes livianos con el ESB utilizando servicios REST basada en el protocolo HTTP. Para el efecto se dispone de una pasarela que convierte los mensajes ESB, cuyo contenido está codificado en RDF (Resource Description Framework) según la ontología OWL (Ontology Web Language), a requerimientos HTTP y viceversa. Este trabajo lo han realizado dentro del marco del proyecto europeo ARUM.

Los trabajos mencionados evidencian la tendencia actual para incorporar tecnologías semánticas en los procesos de integración de fuentes de datos heterogéneas, en el presente trabajo se pretende lograr una arquitectura simple, basada en estándares, con la finalidad de que los resultados de su implementación se puedan alcanzar en un corto plazo.

3. ESCENARIO

Para la elaboración de la propuesta se realizó el análisis de las fuentes de datos de una institución de educación superior que cuenta con alrededor de dieciséis fuentes de datos, de donde se extrajo un caso típico que generalmente se repite en otras organizaciones, como es la gestión de recursos humanos. Para fines demostrativos de los problemas de integración se construyeron dos aplicaciones prototipo

que gestionan la entidad “persona” la cual se emplea en varias fuentes de datos en la organización analizada.

El patrón de desarrollo de software seguido para la implementación de las aplicaciones prototipo es MVC (Modelo Vista Controlador), el cual se fundamenta en separar los datos, la interfaz del usuario y la lógica interna. Es comúnmente usado en aplicaciones Web, donde la vista es la página HTML, el modelo es el Sistema de Gestión de Base de Datos y la lógica interna, y el controlador es el responsable de recibir los eventos y darles solución. Además, cabe indicar que se ha incorporado una capa adicional, la cual contiene servicios Web que permitirán el acceso a los procesos de las diferentes aplicaciones como se muestra en la Figura 2 y que a continuación se describen.

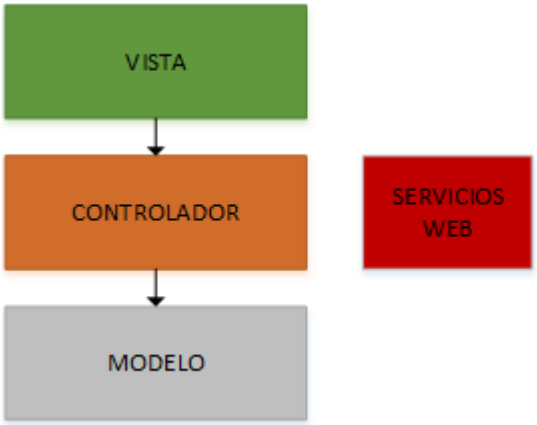


Figura 2. MVC: Arquitectura del prototipo.

3.1. Vista

El objetivo de la interface Web implementada permite soportar el ingreso de datos en la aplicación prototipo. En este caso los datos corresponden a una persona y su información de contacto. Para que el prototipo refleje los problemas reales de integración en las organizaciones se aplican diferentes tecnologías en su implementación, entre ellas están IDEs de desarrollo, base de datos, plataformas de servidores, etc.

3.2. Capa de servicios Web

Es la capa que actúa junto al controlador de aplicación y para cumplir sus propósitos se ha implementado servicios Web que reciben las peticiones del usuario y registran la información correspondiente en el modelo de datos. Los servicios implementados permiten los métodos necesarios para ejecutar las operaciones de ingreso, actualización, y listado de los objetos. Para el caso de estos prototipos se han definido métodos que permiten tratar con la capa de datos, de esta manera se puede gestionar los datos a partir de la fuente. Los métodos definidos son los siguientes: create, edit, destroy, changeList y updateList.

En la implementación de los servicios Web se ha aplicado estándares como: 1) WSDL (Web Services Description Language), que está basada en XML y permite describir la interfaz pública a los servicios Web, así como la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo; 2) SOAP (Simple Object Access Protocol): Es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

3.3. Capa de lógica de negocio

Esta capa es el núcleo de la aplicación. El objetivo de esta capa en el prototipo es que toda la lógica de la aplicación esté bien localizada y no integrada con los objetos de las otras capas. Los métodos que se han creado son: créate, update, destroy, changeList, updateList, cada uno tiene su operación correspondiente en el servicio Web para la interacción.

3.4. Modelo de datos

El modelo de datos usado en el prototipo tiene fines de demostración únicamente, se crearon modelos que representan a una persona con diferentes características, es decir, cada prototipo tiene su propio modelo acerca de una persona, en el primer caso se manejan tres campos (id, nombres, apellidos), y, en el segundo caso, cinco campos (id, nombres, apellidos, fecha de nacimiento, dirección). Sobre estos modelos de datos se describirán los problemas de integración que se tratan en este trabajo.

4. PROPUESTAS DE INTEGRACIÓN

4.1. Integración utilizando un bus de servicios

Esta solución hace uso de un ESB (Menge, 2007) mediante la aplicación del patrón EAI (Enterprise Application Integration) (Linthicum, 2000) a una arquitectura orientada a servicios (SOA). Una solución EAI consiste en comunicar las diferentes aplicaciones mediante conectores, tanto dentro de la organización como fuera de ella. Una de las ventajas de usar un ESB, es que este posibilita la comunicación entre fuentes de datos independiente del protocolo usado. Es decir, se convierte en una pasarela que se encarga de traducir de un lenguaje a otro. El lenguaje usado en el ESB es XML, lo cual facilita el intercambio de mensajes. El ESB es una infraestructura de mensajería y comunicación construida sobre la arquitectura orientada a servicios (SOA), sirve como un middleware que facilita la interoperabilidad de los servicios y aplicaciones heterogéneas dentro de las organizaciones. Técnicamente, el ESB por si solo permite la articulación de sistemas interactivos y permite distribuir la lógica de negocio de una solución en módulos incrementales, manteniendo su propio control local y la autonomía.

En esta propuesta, los servicios web no interactúan directamente, éstos deben ser registrados en el ESB, y su invocación se realiza a través de un conector de servicios SOAP. Internamente en el ESB se definen rutas utilizando diferentes patrones de integración que permite definir un modelo virtual de integración. Cabe indicar, que cuando se necesite integrar una nueva aplicación, se tienen que crear los servicios web correspondientes para registrarlos dentro del ESB. Si en lugar de una nueva aplicación es una base de datos (BD), se debe registrar la BD en el ESB y utilizar un componente de acceso exclusivo para este tipo de fuentes. En ambos casos, el conocimiento acerca del modelo de datos de cada fuente es primordial, puesto que las rutas que se definen en el ESB están basadas en las relaciones que puedan existir entre las entidades de las fuentes de datos.

4.2. Integración utilizando un bus de servicios más tecnologías semánticas

En esta subsección se introduce el concepto de acceso uniforme de datos usado para la integración de fuentes de datos aplicando herramientas basadas en tecnologías semánticas dentro de un ESB. Uno de los factores considerados para optar por una propuesta semántica de integración es que esta solución permite afrontar uno de los retos claves en la integración de datos de distintas fuentes, la heterogeneidad semántica (introducida en la Sección 2.1). Además, este enfoque permitirá definir modelos de integración tanto virtuales como materializados.

Este planteamiento tiene la limitante de ser exclusivamente virtual, es decir, no se mantiene un repositorio común de datos. Si una aplicación esta fuera de línea, puede ocasionar que las rutas definidas que permiten tratar con las fuentes de datos no se ejecuten, razón por la se plantea adicionalmente que el modelo posibilite el funcionamiento en línea como fuera de línea, para el efecto se contempla la creación de un almacén de datos semántico centralizado. A continuación, se describe los componentes principales del prototipo semántico implementado.

1) Ontologías como solución a la Integración Semántica: Se ha adoptado la arquitectura de referencia ANSI/SPARC de tres capas (esquema físico, esquema conceptual y vistas) para que soporte semántica. En esta versión adaptada, se establecen tres capas análogas: documentos, esquema y ontología. Esta división en capas permite separar y solucionar los problemas existentes en todo el proceso de la integración semántica.

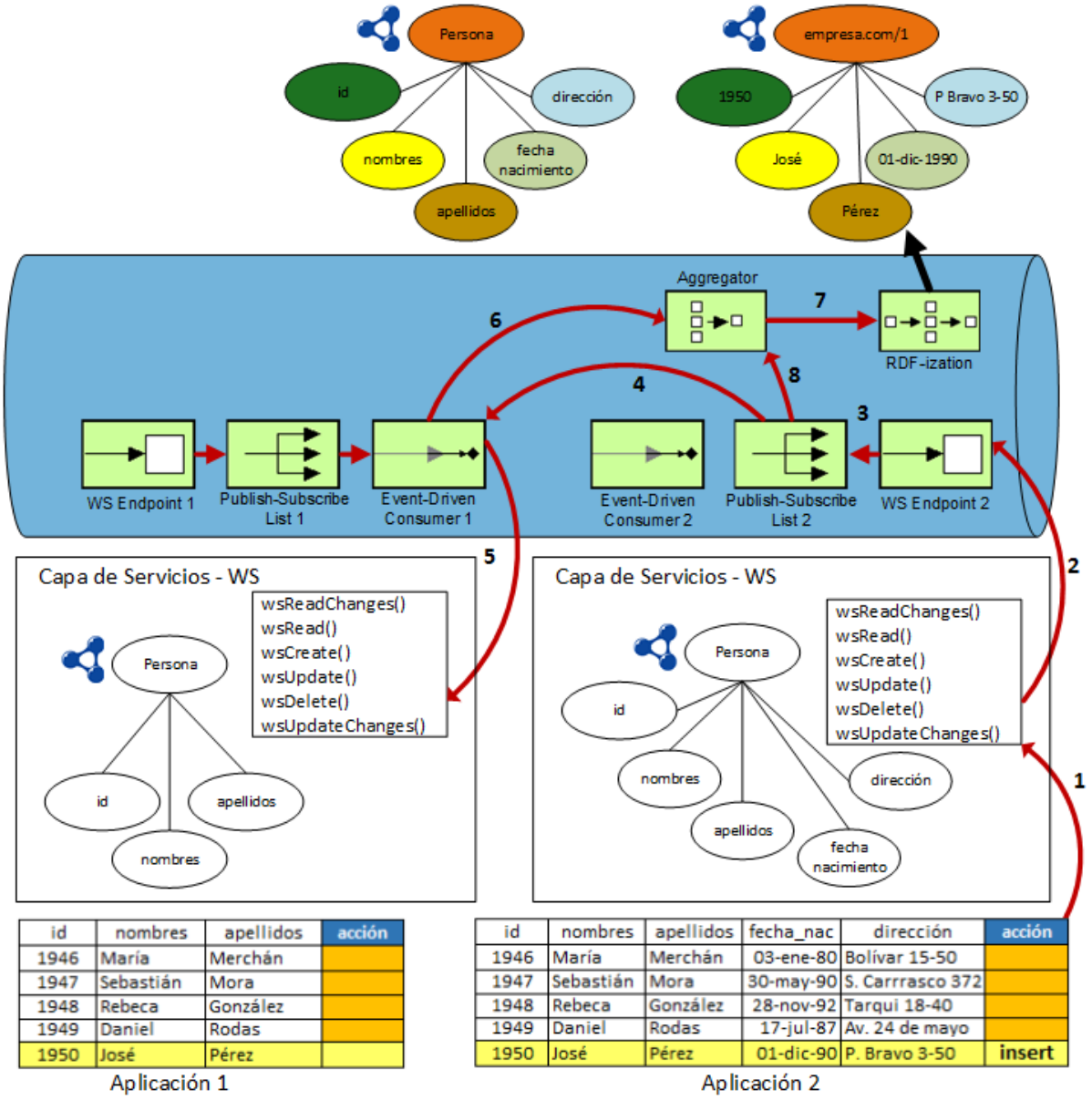
- Documentos: La capa de documentos contiene los datos que se pretende integrar, en su formato nativo o bien se puede utilizar un recubridor que efectué la conversión necesaria hacia documentos XML, RDF u OWL.
- Esquema: La capa de esquema describe la estructura global de los documentos que componen la capa inferior. Se tratará de descripciones mediante XML Schema, RDF u OWL.
- Ontología: La capa de ontología proporciona una visión semánticamente coherente de la información mediante el uso de ontologías que describan el dominio del sistema. El usuario siempre interactúa con esta capa, que proporciona una visión simplificada y de alto nivel, ocultando la heterogeneidad del sistema subyacente. En este caso la Ontología se construye bajo el lenguaje de ontologías OWL.

En la literatura se puede encontrar varios autores que afirman que las ontologías aparecen como solución potencial para resolver el problema de la integración semántica de los datos. Esta propuesta es implementada en el prototipo y los detalles de la misma son explicados en las siguientes secciones.

2) Arquitectura del Prototipo: La plataforma de integración que se propone debe ser fácil de usar, administrar e integrar en un entorno existente. La solución abarca una infraestructura semánticamente enriquecida orientada a servicios, que incluye un ESB, en este caso semántico, para el control, manejo y transformación de los mensajes. La arquitectura del prototipo propuesto fue diseñada de acuerdo con los principios de SOA como altamente modular y extensible. El proceso de creación de la plataforma semántica se compone de las siguientes etapas:
   - Metodología de creación de ontologías: Durante esta etapa se usó la metodología propuesta dentro del proyecto NeOn (Suárez Figueroa, Gómez-Pérez, & Fernández-López, 2012), la cual permite la creación de recursos ontológicos partiendo de vocabularios existentes y similares al dominio de aplicación. Se debe destacar que como primer paso se busca recursos similares, accediendo a repositorios disponibles en la Web. En caso de que estos recursos no sean adecuados para los propósitos del modelo a implementar, se procede a la etapa dos de la Metodología. Esta segunda etapa propone buscar recursos no ontológicos como bases de datos, catálogos, etc., para la construcción de la ontología. Para el escenario descrito se han definido dos modelos ontológicos para representar a cada una de las fuentes de datos usadas por los prototipos. Además, se define un modelo ontológico global que permita mantener una visión homogénea y unificada de toda la organización. La estrategia de mantener modelos locales de cada fuente, y un modelo global de toda la organización, permite tener una visión única de todos los componentes desde la perspectiva del usuario y deja establecidas las bases para generar nuevas aplicaciones basadas en el modelo global.
   - Gestión de ontologías: Una vez definidas las ontologías, durante esta etapa se procede a revisar los conceptos y propiedades que conforman cada prototipo de forma que permitan el intercambio adecuado de conocimiento. En este caso se crearon componentes dentro del ESB que permiten manipular las ontologías, es decir, realizar operaciones de inserción, actualización, borrado y listado de entidades. Estos componentes creados tienen una estrecha relación con los servicios web descritos anteriormente, puesto que parte de los modelos ontológicos circulan a través de las rutas creadas dentro del ESB.
3) Relación entre ontologías: Finalizada la definición de los conceptos y propiedades de cada ontología (locales y global), se procede a relacionar los recursos ontológicos, es decir, se genera los mapeos entre las entidades del modelo global y los modelos locales.
4) Definición de las instancias: A fin de poder hacer uso de los recursos ontológicos creados, en este paso se procede a crear las instancias para cada concepto.
5) Definición de Rutas: Las rutas definidas permite leer información de los servicios Web creados en los prototipos, esta información es transformada y transportada a diferentes componentes formando un flujo de información por el cual circulan los datos de diferentes aplicaciones. Estas rutas están basadas en los patrones de integración (Hohpe & Woolf, 2002), los cuales permiten mantener un estándar de extracción, transformación de los datos de los servicios Web y cargar en el repositorio semántico.

4.3. Modelo de comunicación

En la Figura 3 se muestra el proceso completo de inserción/actualización de información utilizando un ESB, así como los diferentes componentes que interactúan. Para clarificar el ejemplo, se ha definido un problema de integración con los dos prototipos definidos en el escenario e implementados utilizando MVC. Se asume que se tiene la entidad persona en los dos prototipos, en el primer caso existen 3 campos (id, nombres, apellidos) y, en el segundo caso, 5 campos (id, nombres, apellidos, fecha de nacimiento, dirección). Como modelo global se tiene un modelo ontológico con los conceptos (id, nombres, apellidos, fecha de nacimiento, dirección).



id	nombres	apellidos	acción
1946	María	Merchán	
1947	Sebastián	Mora	
1948	Rebeca	González	
1949	Daniel	Rodas	
1950	José	Pérez	

id	nombres	apellidos	fecha_nac	dirección	acción
1946	María	Merchán	03-ene-80	Bolívar 15-50	
1947	Sebastián	Mora	30-may-90	S. Carrrasco 372	
1948	Rebeca	González	28-nov-92	Tarqui 18-40	
1949	Daniel	Rodas	17-jul-87	Av. 24 de mayo	
1950	José	Pérez	01-dic-90	P. Bravo 3-50	insert

Figura 3. Modelo de integración aplicando tecnologías semánticas.

1) El proceso inicia con la inserción de un nuevo registro en la aplicación dos (1950,'José','Pérez','01/04/1990','P. Bravo 3-50', insert), puesto que las fuentes de datos son controladas, se ha creado un campo más a nivel de la BD (acción), la que permite conocer la acción a realizar sobre el registro. En este caso indica inserción de datos. En la capa de servicios Web de la aplicación dos existen diferentes métodos que permiten manipular los registros de la base de datos.

2) En el ESB existe un componente (WS Endpoint 2) que cada cierto tiempo accede a los servicios Web de la aplicación dos en busca de registros nuevos, actualizados o borrados. En este caso detecta que existe un registro nuevo (1950,'José','Pérez','01/04/1990','P. Bravo 3-50', insert) que fue detectado a través del método wsReadChanges.
3) El nuevo registro se envía a través de los servicios web de la aplicación uno (wsCreate), esto permite almacenar el registro creado en la aplicación dos en la base de datos de la aplicación uno.
4) Simultáneamente con el envío de los datos a la aplicación uno, un componente envía el registro del repositorio local de la aplicación uno a un componente temporal de agregación (Aggregator).
5) En este paso, los registros alojados en el componente temporal de agregación son transformados a RDF utilizando el modelo global (RDF-ization). Permitiendo mantener una base de datos semántica de todos los registros de la organización.
6) Simultáneamente con el registro de los datos en el componente Publish-Subscribe List 2, se envía el registro a un componente de agregación temporal.

El proceso de creación/actualización/borrado de registros en las diferentes fuentes de datos se gestiona de la misma manera, con ligeras variaciones en el flujo de datos. Como se puede observar el proceso es generalista, permitiendo la agregación de nuevas fuentes de datos de forma prácticamente instantánea. El proceso más complejo es la detección de modelo ontológico local de la nueva aplicación a integrar, puesto que los componentes son generales para cualquier aplicación.

5. ARQUITECTURA RECOMENDADA

Típicamente, las implantaciones de SOA son concebidas como proyectos a largo plazo, comúnmente, el tiempo de materialización del beneficio y la obtención de retorno sobre la inversión se dilatan en término de años. Sin embargo, con los procedimientos expuestos en la Sección 4, queda demostrada la factibilidad teórica y tecnológica de una arquitectura de integración rápida y con un esfuerzo no excesivo, siendo estas características clave para ganar la aceptación de los actores involucrados. La definición de esta arquitectura de integración de fuentes de datos se desarrolló siguiendo métodos actuales, así como tecnologías de punta, tanto en el ámbito de las tecnologías semánticas así como con las tecnologías ESB.

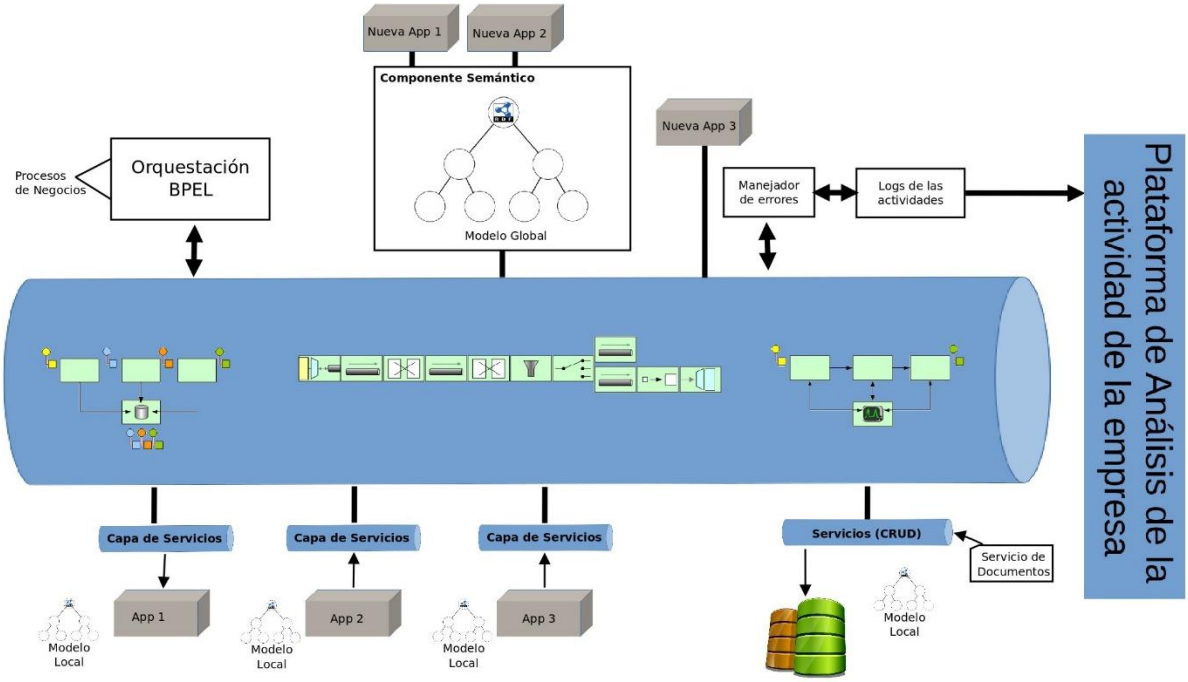


Figura 4. Arquitectura recomendada.

Como resultado final, en la Figura 4 se muestra la propuesta de integración, en donde el ESB posibilita la integración de fuentes de datos heterogéneas, la integración de cada fuente de datos implica el desarrollo de interfaces basadas en servicios Web que deben registrarse y enlazarse a las rutas definidas en el ESB, además se debe determinar el modelo ontológico local que represente las entidades y relaciones de la fuente de datos. Otro elemento que genera valor a la propuesta es el mantenimiento del modelo ontológico global, el cual sirve como punto de partida para la construcción de nuevas aplicaciones.

La importancia de esta arquitectura radica en que permite tener una visión futurista en cuanto al desarrollo o adquisición de nuevas aplicaciones. Teniendo en cuenta la arquitectura que se recomienda, ESB + semántica, esta quedaría preparada para evolucionar hacia una verdadera integración a nivel global, utilizando los principios que marcan las tecnologías de la Web Semántica. Además, permitirá mantener un almacén de datos semánticos centralizado en donde se pueda realizar diferentes tipos de análisis de información o extracción de conocimiento, utilizando técnicas de minería de datos.

Tomando en consideración esta arquitectura se puede planear la creación de nuevas aplicaciones basadas en la ontología global o directamente a través de las rutas definidas en el ESB. Por otra parte, se puede pensar en aplicación BPM o BPEL que pueden actuar sobre el ESB, afectando a todo el modelo de integración descrito en la propuesta.

6. CONCLUSIONES

En este trabajo, se ha mostrado como la semántica juega cada día un papel más importante a la hora de satisfacer una determinada necesidad de información. Particularizando el problema de la semántica a la integración de datos provenientes de distintas fuentes heterogéneas entre sí, se ha visto como esta propuesta implica necesariamente el uso de algunas ontologías, tanto para modelar las aplicaciones locales, así como una ontología que permita modelar a la organización. Además, es necesario establecer las relaciones semánticas entre los modelos locales y el global. El modelo ontológico ideal para trabajar, deber ser aquel que sea consensuado por una mayoría, sea reusable y favorezca la usabilidad en el mundo real. Estas consideraciones han sido recogidas en la propuesta de integración que permita un acceso uniforme de los datos provenientes de diferentes fuentes.

Por otra parte, se ha podido evidenciar que la infraestructura provista por el ESB es sin lugar a duda la más adecuada para llevar procesos de integración de fuentes de datos. La combinación de ESB más semántica se presenta por tanto como una solución viable para obtener la integración efectiva de fuentes de datos en las organizaciones. Es necesario manifestar que la implementación desarrollada es un prototipo, que busca mostrar la viabilidad de este enfoque.

El trabajo futuro se orientará hacia el descubrimiento automático de los modelos que representen a una fuente de datos. Estos modelos permitirán, a largo plazo, un proceso de integración automático. Por otra parte, para demostrar la viabilidad de la propuesta, se planea realizar una experimentación sobre fuentes de datos en un entorno real. Finalmente, se pretende utilizar técnicas de procesamiento de lenguaje natural para establecer las relaciones entre los modelos ontológicos.

AGRADECIMIENTO

Los autores expresan su agradecimiento al programa de Maestría en Gestión Estratégica de Tecnologías de la Información, Facultad de Ingeniería, Universidad de Cuenca.

REFERENCIAS

Bernstein, P. A., Haas, L. M. (2008). Information integration in the enterprise. *Communications of the ACM*, *51*(9), 72-79. doi:10.1145/1378727.1378745

Beyer, M. A., Thoo, E., Zaidi, E., Greenwald, R. (2016). *Gartner's magic quadrant for data integration tools*. Recuperado de https://www.gartner.com/home

Dittrich, K. R., Jonscher, D. (1999). *All together now: Towards integrating the world's information systems*. In: Advances in Databases and Multimedia for the New Century - A Swiss Japanese Perspective, 7 p.

Harcuba, O., Vrba, P. (2015). *Unified REST API for supporting the semantic integration in the ESB-based architecture*. In: IEEE International Conference on Industrial Technology (ICIT), Sevilla, pp. 3000-3005.

Hohpe, G., Woolf, B. (2002). *Enterprise integration patterns*. In: 9th Conference on Pattern Language of Programs, 19 p.

Jin, J. Y., Lv, F. Z., Xiang Z. Q. (2014). Using semantic technology to enhance ESB capabilities, *Advanced Materials Research*, *849*, 298-301.

Lenzerini, M. (2002). *Data integration: A theoretical perspective*. Proceedings of the ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. pp. 233-246.

Linthicum, D. S. (2000). *Enterprise application integration*. Reading, MA: Addison-Wesley Longman.

Menge, F. (2007). *Enterprise service bus*. In: Free and Opensource Software Conference. 16 p.

Roa-Valverde, A., Aldana-Montes, J. (2008). *Extending ESB for semantic web services understanding*. In: Meersman, R., Tari, Z., Herrero, P. (Eds.). On the Move to Meaningful Internet Systems, OTM 2008 Workshops, Vol. 5333 of Lecture Notes in Computer Science, pp. 957-964, Springer Berlin Heidelberg.

Schratzenstaller, W. M. K., Baldo, F., Rabelo, R. J. (2016). *Semantic integration via enterprise service bus in virtual organization breeding environments.* In: Nguyen N. T., Trawiński, B., Fujita, H., Hong, T. P. (Eds.). Intelligent Information and Database Systems. ACIIDS 2016. Lecture Notes in Computer Science, Vol 9622. Springer, Berlin, Heidelberg

Shi, K., Gao, F., Xu, Q., Xu, G. (2014). *Integration framework with semantic aspect of heterogeneous system based on ontology and esb*. In: Control and Decision Conference (2014 CCDC), The 26th Chinese, pp. 4143-4148.

Suárez Figueroa, M. C., Gómez-Pérez, A., Fernández-López, M. (2012). *The NeOn methodology for ontology engineering*. In: Ontology Engineering in a Networked World, pp. 934. Springer Berlin Heidelberg.

Tsierkezos, S. A. (2010). *Comparing data integration algorithms*. Doctoral dissertation, Thesis abstract, 29 p. University of Manchester, Manchester, UK. Disponible en https://studentnet.cs.manchester.ac.uk/resources/library/thesis_abstracts/BkgdReportsMSc10/Tsierkezos-Sebastian.pdf