

Enfoque para la aplicación de Web Semánticas en Sistemas Centrados en el Usuario

Fernando Sanabria¹, Jorge Salvador Ierache^{1,2}

1. Instituto de Sistemas Inteligentes y Enseñanza Experimental de la Robótica. Facultad de Informática, Ciencias de la Comunicación y Técnicas Especiales. Universidad de Morón
2. Laboratorio de Sistemas Inteligentes. Facultad de Ingeniería. Universidad de Buenos Aires
fersanabria@yahoo.com.ar, jierache@unimoron.edu.ar

RESUMEN

Los servicios basados en la Web ya se han convertido en una herramienta más dentro de la vida de la mayoría de las personas. En la actualidad, por ejemplo, se utiliza la Web para pagar servicios, comprar artículos o escuchar programas de radio. Todo indica que esto seguirá expandiéndose y que cada vez habrá más aplicaciones para que los usuarios puedan acceder a más servicios. La idea de dar un enfoque Semántico a la Web, enriquece enormemente este entorno ya que habilita a las computadoras a entender el contenido de la información que están intercambiando. Semantic Web Services a su vez, permite que diferentes servicios desplegados en la Web puedan interactuar de manera automática a un nivel semántico, basándose en las descripciones de sus interfaces usando ontologías.

Tomando de base una tecnología emergente y tan prometedora como la Web Semántica, este trabajo analiza los aspectos más relevantes para lograr un enfoque de Web Semántica aplicada en Sistemas Centrados en el Usuario y propone una solución para uno de los elementos fundamentales de este enfoque, como es un Servidor de Modelos de Usuario (SMU).

1. Introducción

Es un objetivo de la comunidad informática en general que las aplicaciones puedan conformarse de manera dinámica para responder de la forma más precisa posible a los requerimientos de un usuario en un momento dado. Este es un objetivo, si bien realizable, un tanto ambicioso para poder ser alcanzado en el corto plazo. Además de esto la Web tiene una población de usuarios creciente y altamente heterogénea, y las empresas en general continúan en la tendencia de reemplazar sus métodos tradicionales de comunicación con sus clientes por aplicaciones basadas en la Web.

Basándose en anotaciones semánticas tomadas de Semantic Web [1], Semantic Web Services [2] es un enfoque de middleware que apunta a posibilitar que diferentes agentes o aplicaciones convencionales puedan conformar soluciones de manera dinámica y circunstancial

para responder a un requerimiento específico. Esto se lograría mediante ciertas facilidades como descubrimiento, interacción ó composición de Web Services de manera dinámica [6]. Por otro lado, desde hace ya más de 2 décadas [5], se viene trabajando sobre diferentes formas de modelar las características del usuario para que puedan ser bien conocidas por las aplicaciones con las que éste interactúa. Sumar el potencial de estos dos aspectos en los que se viene trabajando en el último tiempo (SWS y Modelos de Usuario), y hacerlos parte de un mismo enfoque es el objetivo propuesto en este trabajo. Esto implicaría dos ventajas importantes: poder realizar composición de servicios de manera totalmente dinámica y evitar al usuario realizar tareas operativas, rutinarias o reiterativas, ya que las aplicaciones responderían a sus requerimientos tomando en cuenta sus preferencias de manera anticipada y convirtiéndolo en algo “transparente” para el mismo. Además basándose en el conocimiento que

o

incrementalmente irán ganando sobre el usuario, en diferentes situaciones podrán hacer propuestas de manera activa a éste asistiéndolo en su interacción con la aplicación y en sus decisiones.

En la próxima sección se presenta una introducción a SWS y Modelos de usuario, por ser estos los pilares del enfoque desarrollado en este trabajo. En la sección 3 se presentan los aspectos a solucionar para proveer un contexto. Este contexto es necesario para entender claramente el porqué del enfoque abordado. En la sección 4 contiene la visión general del SMU y los beneficios que aportaría a la comunidad de Internet. Finalmente las conclusiones.

2. Semantic Web Services y Modelos de Usuario

2.1 SWS

Siguiendo a [4], mientras que el advenimiento de las arquitecturas SOA basadas en Web Services han presentado una buena base con respecto a middleware basado en la Web, todavía hay algunos aspectos que no han sido cubiertos y que quizás no puedan ser solucionados por los “ingredientes” clásicos de Web Services, tales como SOAP, Web Service Description Language (WSDL), Universal Description, Discovery and Integration (UDDI). Crear aplicaciones de software requiere tiempo y esfuerzo de especialistas de IT calificados. Aún mientras practicas de ingeniería de software bien probadas son aplicadas de forma extensiva, existe todavía un cierto vacío entre las intenciones y deseos de los usuarios y las funcionalidades ofrecidas por las aplicaciones de software desarrolladas [3]. Es un deseo aún la posibilidad de que, por ejemplo, agentes puedan, acorde con requerimientos del usuario o el negocio, configurar aplicaciones “on-the-fly” basándose en componentes y servicios disponibles en la Web. Para hacer uso de un Web Service, un agente necesita una descripción del servicio interpretable por un programa de computadora y todos los instrumentos necesarios para poder hacer un “matching” dinámico del Web Service. Si queremos avanzar hacia descripciones más enriquecidas de los Web Services, la Web Semántica es la opción evidente. Como describió [1] “La Web semántica es una extensión de la Web actual en la cual se le da un significado bien definido a la información, posibilitando un trabajo cooperativo mejorado entre personas y computadoras”. Un Semantic Web Service describe sus propiedades y capacidades a fin de que aplicaciones de software puedan determinar automáticamente su propósito. Las Ontologías para servicios Web deberían hacerlos entendibles por máquinas y soportar ciertas características fundamentales como descubrimiento de servicios automático, invocación,

composición. En resumen, permitir una interoperabilidad automática para que un conjunto de servicios puedan cumplir un determinado objetivo. De esta manera, entidades de software pueden crear soluciones completamente nuevas, las cuales son parte de colaboraciones de negocio on-demand.

En la Figura 1 se pueden apreciar las diferentes dimensiones en las que se pueden estructurar los componentes de Semantic Web Services. Estas dimensiones son: Actividades, Arquitectura y Ontología de servicio [6].

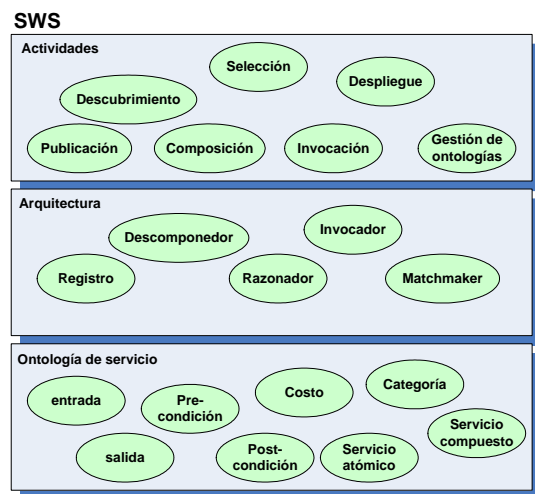


Fig. 1 Dimensiones de SWS

2.2 Modelado del Usuario

El modelado de usuario tiene sus comienzos por fines de la década del 70' cuando Allen, Cohen y Perrault presentan los primeros trabajos basados en esta idea (p. E., Perrault, 1978; Cohen y Perrault, 1979; Allen, 1979). En estos comienzos en el modelado de usuario, el modelado se realizaba en la aplicación misma y no existía una clara distinción entre los componentes que administraban la información del usuario y los componentes que realizaban otras tareas.

A partir de mediados de los años ochenta, tal separación se hizo cada vez más visible (p. e., Kobsa, 1985; Sleeman, 1985; Kass, 1988; Allgayer, 1989), pero no se puso énfasis en la modularización de los componentes de modelado de usuario para permitir que sean reutilizables en nuevos sistemas “user-adaptive”.

Ya para fines de los 90' el valor de la personalización en la Web es altamente reconocido en el área del comercio electrónico (Hof et al., 1998; Allen et al., 1998;

Cooperstein et al., 1999; Hagen et al., 1999). La característica central de los sistemas de modelado de usuario a partir de ese momento fue la modalidad servidor, en la cual estos no se encuentran integrados en las aplicaciones mismas sino que las sirven a través de comunicación entre procesos. De esta manera pueden proveer información a más de una aplicación de manera centralizada.

La personalización, interés fundamental de la comunidad de modelado de usuario, refiere a métodos para ganar algún conocimiento acerca de los usuarios, representados como un modelo de usuario, y explotando ese modelo para ajustar el comportamiento de los sistemas a las necesidades de los individuos. Gracias al aporte de la Web Semántica los modelos de usuario pueden disponer hoy de estándares bien definidos y ontologías, para proveer extensibilidad, flexibilidad, interoperabilidad y reusabilidad.

3. Aspectos a solucionar: “El contexto”

En esta sección se intenta presentar de manera clara el contexto, con todos sus aspectos, que da origen al enfoque desarrollado en el trabajo. Entre estos aspectos está el de un servidor de modelos de usuario. Para introducirnos de manera clara, vamos a describir algunas situaciones factibles de la vida real de un usuario como ejemplo.

Consideremos un usuario planificando y ejecutando un viaje de placer por Miami para sus vacaciones. Lo primero que nuestro usuario deseará hacer son todas las reservas de vuelos, hospedaje y un auto, para lo cual necesitaría realizar las búsquedas, reservas e ingresos de datos reiterados necesarios con todo el trabajo reiterativo y tedioso que esto implica. Pensemos ahora en una aplicación de planificación de viajes centrada en el usuario en la cual el usuario ingresa origen y destino de vuelo, fechas, costos, cómo espera que esté el tiempo en su estadía, comodidades del hotel en el cual le gustaría hospedarse y que tipo de auto desea rentar para moverse por aquella ciudad. Todo esto desde un solo punto de entrada y nuestra aplicación se encargará de hacer la búsqueda con estos requisitos y presentarle al usuario diferentes opciones para que él pueda tomar la decisión final. Una vez que se decide, sólo debe aceptar y se toman automáticamente sus datos personales, número de tarjeta de crédito y otros datos requeridos, de su modelo de usuario digital personal. Luego puede recibir por correo los boletos y otra documentación necesaria o puede retirarla personalmente según haya elegido. Una vez en Miami ya lo estará esperando su auto rentado y su habitación reservada. Se podría seguir describiendo como el usuario, acompañado de su PDA, sigue utilizando servicios personalizados gracias a un servicio de Internet wireless y aplicaciones centradas en su perfil y preferencias.

Con este ejemplo como referencia, se podrá entender mejor la necesidad de los aspectos que se consideran relevantes para un enfoque de Web Semántica en Sistemas Centrados en el Usuario.

3.1 Middleware de Semantic Web Services

Que en la actualidad podamos estar hablando de todas las bondades que tienen los Semantic Web Services y lo veamos como una realidad no tan lejana, es porque muchas instituciones, Universidades, Empresas, Comunidades, están dedicando un gran esfuerzo en investigación y desarrollo. El avance demostrado en este enfoque en el último tiempo hará que en un futuro no tan lejano podamos basarnos en una interoperabilidad estandarizada, sencilla y robusta a niveles semánticos a la hora de diseñar soluciones futuras [2, 6].

3.2 Modelo del usuario

Un modelo es una representación simplificada de una parte de la realidad. Si el objetivo es que las aplicaciones puedan interactuar de una manera más personalizada con los usuarios, éstas deben contar con información acerca de los usuarios para usarla como fuente en la toma de decisiones. Cuanto mayor y más depurada es la información que contiene el modelo del usuario, mejor y más ajustada es la interacción del sistema que lo utiliza con el usuario. Un modelo de usuario mantenido en un servidor al que puedan acceder diferentes “clientes” para consultas y actualizaciones constantes es un elemento fundamental para poder centrar en el usuario las aplicaciones distribuidas en la Web. El modelo del usuario podría actualizarse y consultarse por diferentes mecanismos, y de esta manera crece el conocimiento que se tiene del usuario.

3.3 Semantic Web Services desplegados

Este es un aspecto que escapa de las posibilidades de un planteo técnico de una solución ya que lograr que la Web se plague de Web Services desplegados, aumentando notablemente la disponibilidad, solo dependerá de lo prometedor y tentador que se presente un modelo como el aquí presentado para los fines comerciales. Dándose esta situación, debería ser natural el crecimiento exponencial de una población de Web Services accesibles en Internet. A todas las entidades comerciales, que podrían publicar Web Services (mayor parte de la población), se sumarían los desplegados por instituciones gubernamentales,

universidades, centros de investigación y demás entidades sin fines de lucro.

La cantidad y diversidad de servicios publicados crea un entorno propicio para un mercado electrónico extensivo en Internet. En este entorno tienen lugar aplicaciones centradas en el usuario.

3.4 Aplicaciones para negocios o problemas específicos

Para hacer un uso de los Web Services desplegados en la red en un negocio específico o para resolver un determinado problema, necesitamos una entidad que “orqueste” estos servicios de la manera adecuada para llegar a un objetivo determinado. Para responder en forma personalizada a los objetivos del usuario, debe haber aplicaciones que sean el punto de entrada para los requerimientos de estos y que se encarguen de hacer las composiciones de Web Services que sean necesarias. Estas aplicaciones que llamamos Centradas en el Usuario (C-U) son la “fachada” que el usuario ve y por lo tanto con la única entidad con la que interactúan cuando persiguen un objetivo determinado o un conjunto de objetivos fuertemente relacionados. Puede haber aplicaciones de mayor capacidad de decisión y autonomía implementadas con agentes, otras que tengan procesos corriendo en batch para procesar información que luego presentarán al usuario o también puede haber aplicaciones del estilo Web clásico que hagan uso de algunos Web Services.

Las aplicaciones C-U son una de las partes fundamentales del enfoque orientado al usuario, es en éstas donde está la lógica que administra los demás “recursos” para poder responder a las necesidades del usuario.

Además de estos aspectos fundamentales, existen otros aspectos complementarios que deben tenerse presentes para abarcar la problemática en su totalidad. Estos son el método de cobro en un entorno de comercialización electrónica y las interfaces de usuario adaptables a las características del hardware.

Los Web Services desplegados en Internet, se encuentran allí porque obtienen algún beneficio económico por el servicio que proveen. En los casos en que por el servicio provisto, la empresa no obtiene ningún tipo de rédito económico el beneficio puede obtenerse por el cobro del uso de un Web Service. Un ejemplo de esto es un Web Service que provea información meteorológica. Para poder gestionar el cobro de consumo de Web Services será necesario un mecanismo y estándares bien claros que definan como se realizarían esas operaciones en un entorno Web.

Para lograr que las aplicaciones sean accesibles de igual manera desde cualquier dispositivo y en toda su funcionalidad se debe tomar en cuenta que las

características de accesibilidad de cada dispositivo varían, y en algunos casos ampliamente. Estas diferencias en la interacción con el usuario no deben tener impacto en la lógica de negocio de las aplicaciones sino que deben manejarse de manera transparente tanto para la aplicación como para el dispositivo del usuario.

4. Servidor de Modelos de Usuario (SMU)

4.1 Descripción general

Como definición general podemos decir que un SMU es una entidad que administra, de manera centralizada, los modelos de usuarios de las aplicaciones que lo utilizan. Presentando características básicas como interfaces estandarizadas de acceso para entornos abiertos y administración optimizada de la información utilizando ontologías.

Cada vez en mayor proporción las aplicaciones necesitan adaptarse a usuarios individuales por la actual tendencia a la personalización en la interacción con los mismos. Para lograr esto es necesario disponer de un modelo del usuario con la mayor cantidad de información y lo más precisa posible. Además de esto, ya no podemos pensar en que cada aplicación administrará en forma privada todas las instancias de modelos de sus usuarios porque esto provocaría que los usuarios que utilizan varias aplicaciones, tengan que mantener actualizada su información en cada aplicación. Lo cuál deriva en consecuencias no deseadas como inconsistencia de información del usuario entre las diferentes aplicaciones que éste utiliza, una actualización de datos engorrosa por parte del usuario ya que tendría que actualizar sus datos en cada aplicación por separado, pérdida de control del usuario sobre que información de él tiene cada aplicación, etc.

Esto conlleva a la necesidad de tener un lugar común en el cuál guardar y administrar los modelos de usuarios, y al cual puedan recurrir los desarrolladores de diferentes aplicaciones para soportar gran parte de su lógica en el uso de servicios de este. Esta necesidad nos orienta hacia el desarrollo de un componente de software con características como las siguientes:

- Ser totalmente independiente y auto contenido
- La información del usuario es mantenida de manera centralizada o virtualmente integrada en un repositorio y puesta a disposición de más de una aplicación al mismo tiempo.
- La información de usuario adquirida por una aplicación puede ser usada por otra y viceversa
- Posibilitar la formación de grupos y estereotipos.

- Presentar una interfase de acceso bien clara y definida
- Ser multi-plataforma, permitiendo entre otras cosas su migración con total independencia del hardware.
- Proveer métodos de backup y restore de toda la información mantenida
- Contar con una documentación clara y completa

La arquitectura del SMU se encuentra estructurada en capas que tienen diferentes responsabilidades en la operatoria general. En la figura 2 se muestra una visión abstracta de la composición en capas (delimitadas con líneas punteadas) donde se aprecia como interactúan entre sí y como están compuestas.

Las aplicaciones centradas en el usuario son los clientes principales del SMU, que lo acceden a través de sus interfaces externas. El SMU presenta interfaces de acceso basadas en Web Services, las cuales mediante un archivo WSDL establecen claramente sus parámetros de acceso. El núcleo del SMU es el que implementa el modelo del mismo y la lógica de negocio. Aquí también se realizan los mapeos entre ontologías para obtener esquemas de ontologías más abstractos que sirvan para abastecer los esquemas más específicos. Finalmente está la gestión de la persistencia de los datos la cuál gestiona los esquemas de ontologías y las instancias de estos. Para almacenar los datos utiliza bases de datos relacionales y archivos XML según aplique.

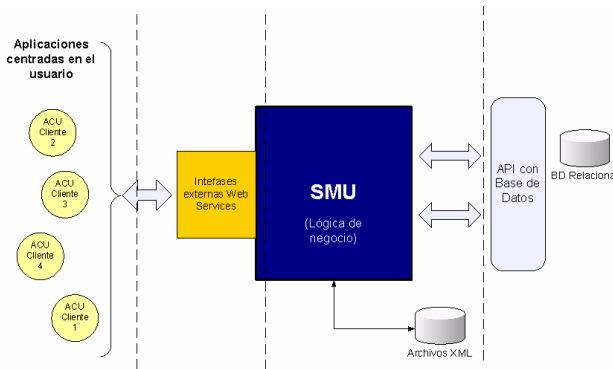


Figura 2. Capas del SMU

Describiendo el SMU en base a sus principales módulos y los componentes que los conforman (Fig. 3), tenemos:

- Modelo

Este módulo concentra toda la lógica del “core” del SMU y actúa como el orquestador de todos los demás componentes para realizar los objetivos finales del SMU. Los servicios este módulo los publica al resto de la arquitectura a través del componente “Servicios del SMU”.

Que lo que hace es publicar las funcionalidades finales del SMU pero internamente.

- Interfase Gráfica de Usuario

El componente fundamental aquí es el que gestiona la GUI de Usuario y que se soporta en las Java Server Pages (JSP) y el framework Struts que implementa el patrón MVC.

- Interfaces externas

Las interfaces hacia el exterior en un entorno Web se implementan en el componente de interfaces Web Services. Este componente utiliza la API Axis de Apache y se publica por medio de un archivo WSDL.

- Gestión de Ontologías

En este módulo se encuentran todos los componentes que realizan algún tratamiento de ontologías. Desde el que realiza un análisis sintáctico de las mismas (parsing), hasta el que las persiste en la base de datos relacional utilizando la API Minerva. El tratamiento de las ontologías en memoria se realiza utilizando la API EODM (EMF Ontology Definition Metamodel). El lenguaje de consultas utilizado es SPARQL[7].

- Validaciones y Seguridad

En este módulo se encuentran todos los artefactos dedicados a la gestión de la seguridad y la validación y verificación de los datos. Dentro de seguridad está la autenticación y autorización de usuarios, sumado a la gestión de las sesiones.

- Registros y notificaciones

El proceso de registración de clientes y usuarios está soportado por el componente de Gestión de registros. A su vez este componente utiliza otro componente para la gestión del mailing.

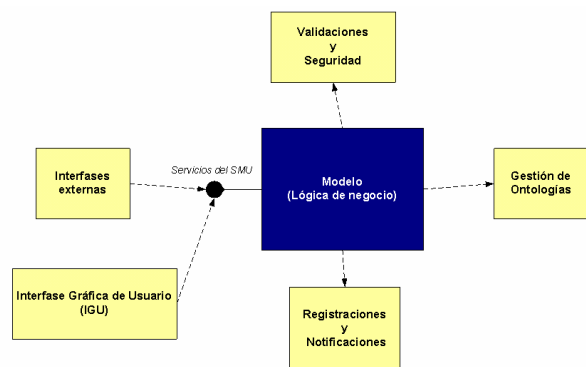


Figura 3. SMU en módulos.

4.2 Beneficios aportados a la comunidad de Internet

El objetivo final del SMU y por el cuál podría llegar a ser concebido, es el de aportar beneficios importantes a la comunidad de Internet. Internet actualmente presenta un escenario de desarrollo exponencial e intrínsecamente incontrolado. Esa comunidad creciente está poblada de sitios estáticos y aplicaciones dinámicas altamente heterogéneas tanto en sus tecnologías de implementación como en sus fines funcionales. Dentro de este esquema arquitectónico de Internet se presenta una población de usuarios de la misma que también se encuentra en un crecimiento constante y que tiene sus límites en al población mundial total.

Estas dos variables principales se suman al interés de los stakeholders del negocio de Internet que intentan desarrollar cada vez en mayor medida una relación con sus clientes a través de este medio masivo y hacerlo continuamente de la manera más personalizada posible.

Esta situación y entorno actual de Internet obligan a definir ciertos elementos que faciliten a la organización de la operatoria general. Para aportar a este fin el SMU propone un servidor de modelos de usuario para sugerir lineamientos de cómo gestionar la información de los usuarios de una manera centralizada.

Las principales ventajas que provee el SMU a los stakeholders de la Web son:

4.2.1 Conocer datos del usuario sin pedírselos.

Una de las cosas más tediosas hoy en día para los usuarios de Internet es la necesidad de ingresar sus datos repetidas veces en diferentes aplicaciones con las que desean realizar alguna acción. La restricción de que los sitios necesitan conocer cierta información del usuario para su negocio y los usuarios desean trabajar con ese sitio hace parecer ineludible la necesidad de que los usuarios tengan que ingresar repetidas veces sus datos.

Para solucionar este problema, la propuesta del SMU es que las aplicaciones registradas como clientes de éste puedan conocer información del usuario sin que necesariamente el usuario la ingrese. Esto se logra gracias a que es probable que el SMU ya tenga en su base de datos la información que la aplicación cliente está necesitando y de esa manera evita al usuario de tener que volver a ingresarla. Esa información que el SMU ya tiene del usuario fue ingresada por él en otro momento, mientras trabajaba con otra aplicación cliente del SMU. Todas las aplicaciones registradas como clientes del SMU deben presentar un logo en sus páginas que las identifiquen como "SMU Supported". De esta manera el usuario puede saber que la aplicación puede acceder a sus datos interactuando con el SMU.

4.2.2 Confianza de las aplicaciones de que los datos del usuario son reales.

Es muy común actualmente que cuando un determinado sitio Web solicita ciertos datos del usuario, antes de permitirle descargar algún archivo o realizar alguna operación, el usuario ingrese datos falsos para pasar ese paso y lograr su objetivo. Sin dudas los datos que el sitio está solicitando son de valor para el negocio de éste y de saber que los datos recibidos del usuario son falsos no permitiría que este continúe con el paso siguiente. Pero hasta el momento los sitios no tienen otra manera de hacerlo.

Con el uso del SMU los sitios registrados como clientes pueden tomar los datos del usuario desde allí. Esto implica la siguiente afirmación y consecuentes alternativas:

Afirmación

Ahora tanto el sitio como el usuario saben que en caso de que el usuario acepte y se loguee en el SMU, los datos que se obtengan de él serán verdaderos.

Alternativas

Sabiendo que los datos que tomará el sitio son verdaderos el usuario puede decidir si continuará o no con el proceso de registración. Esto también dependerá de si el sitio le permite o no modificar los datos. El SMU puede decidir si al momento de presentar los datos al usuario le permitirá que los modifique o no.

Si aún obteniendo los datos del SMU al sitio le siguen faltando datos por completar, al menos el usuario sabrá que los datos que ingresen serán actualizados en el SMU y por lo tanto dudará de ingresar datos falsos.

En resumen el SMU plantea una nueva manera de gestionar la información de los usuarios en la Web, marcando pautas mucho más claras que le son altamente beneficiosas a los usuarios y aún en mayor medida a las aplicaciones desplegadas.

4.2.3 Hacer que los usuarios perciban confiabilidad en las aplicaciones que utilizan el SMU.

Si la imagen del SMU dentro de la Web comienza a valorarse y expandirse, el hecho de que una aplicación sea soportada por SMU y se identifique como tal, podría ser percibido por los usuarios como un atributo positivo de la aplicación. Al establecerse un estándar en como se gestionan los datos referentes a los usuarios, la simple identificación como una aplicación que trabaja con el SMU generaría confianza. Esto haría que la mayoría de las aplicaciones traten de convergir hacia la misma tendencia.

4.2.4 Interacción facilitada entre aplicaciones con respecto a la información del usuario.

La interacción entre aplicaciones Web que necesitan intercambiar información relacionada con el usuario sería mucho más fácil si ambas están soportadas por SMU. Esto básicamente porque tienen la seguridad de que trabajan con la misma información del usuario. Totalmente distinto es cuando cada una tiene su propio esquema e información del usuario. Un ejemplo concreto puede ser la compra de un producto y el envío. Si el usuario está registrado en la empresa que vende el producto y en la empresa que realiza el envío, la aplicación que vendió el producto solo tiene que indicarle a la que lo entrega cuál es el usuario destinatario y nada más.

5. Conclusiones

Se identificó la necesidad de ir hacia un enfoque que integre diferentes temas de estudio, que actualmente se están desarrollando en la comunidad informática, con el fin de lograr aplicaciones aptas para entornos abiertos y que se centren en las características del usuario. Entre estos temas en desarrollo se destacan como fundamentales la Web Semántica y los Modelos de usuario. Luego de identificarse el contexto del problema con todos sus elementos e interrelaciones, se continúa hacia el diseño de la arquitectura para un Servidor de Modelos de Usuario.

El fin último de este desarrollo son los beneficios que puede traer a la comunidad de Internet, en ciertas problemáticas que experimenta actualmente a causa de su desarrollo exponencial.

6. Referencias

- [1] T. Berners-Lee, et al. The Semantic Web, Scientific American, May 2001.
- [2] Semantic Web Service Initiative (SWSI). www.swsi.org
- [3] J. Domingue, L. Cabral, et al. "A Position statement on IRS-III: A Comprehensive Approach to Creating and Using Semantic Web Services", 2005.
- [4] A. Svirskas, M. Wilson, et al. "Towards an Efficient, Reliable and Collaborative Web: from Distributed Computing to Semantic Description, Composition and Matchmaking of Services", 2005.
- [5] A. Kobsa, Generic User Modeling Systems, 2000.
- [6] L. Cabral, J. Domingue, et al. Approaches to Semantic Web Services: An Overview and Comparisons, 2004.
- [7] SPARQL Query Language for RDF. <http://www.w3.org/TR/rdf-sparql-query/>
- [8] Data, Information, and Process Integration with Semantic Web Services (DIP). dip.semanticweb.org/index.html
- [9] Katia P. Sycara. Multiagent Systems. 1998.