

Social Pipes: Enriqueciendo contenidos web mediante la inclusión no intrusiva de aspectos de redes sociales

Alicia Díaz
LIFIA, F.I., UNLP
La Plata, Argentina
+54 221 4228252

alicia.diaz@
lifia.info.unlp.edu.ar

Alejandro Fernández
LIFIA, F.I., UNLP
La Plata, Argentina
+54 221 4228252

alejandro.fernandez@li
fia.info.unlp.edu.ar

Eduardo Lucas
LIFIA, F.I., UNLP
La Plata, Argentina
+54 221 4228252

eduardo.lucas@
lifia.info.unlp.edu.ar

Danilo Zecchin
LIFIA, F.I., UNLP
La Plata, Argentina
+54 221 4228252

danilo.zecchin@
lifia.info.unlp.edu.ar

RESUMEN

Un sitio web enriquecido socialmente, es un sitio que publica contenidos y provee (integra) mecanismos para vincularlos con las redes sociales. Si bien existen tecnologías que asisten en la construcción de sitios enriquecidos, las mismas tienen importantes limitaciones. Social Pipes es un servicio para transformar sitios web tradicionales en sitios web enriquecidos socialmente. Su estructura de diseño tipo framework permite que el mismo sea extendido para agregar nuevas acciones de socialización y nuevos proveedores de redes sociales.

Descriptor de Temas y Categorías

D.2.11 [Software Architectures]: patterns (pipeline), SOA.

Términos Generales

Algoritmos, Documentación, Diseño, Lenguajes.

Palabras clave

Redes sociales, Enriquecer, No intrusivo, WWW.

1. INTRODUCCIÓN

La popularidad de los sitios de redes sociales (SNS) es un reflejo de nuestra necesidad de conectarnos, comunicarnos y expandir nuestras conexiones. A lo largo de la historia el número y la calidad de nuestras conexiones han estado limitado por barreras como la distancia, el tiempo, la cultura. Internet ha derribado muchas de esas barreras permitiéndonos expandir nuestras redes sociales. Los sitios y servicios de redes sociales han originado un importante cambio en la forma en la que nos conectamos y comunicamos on-line. Cambia, por ejemplo, la forma en la que manejamos nuestra libreta de contactos (ahora red de amigos), la forma en la que nos enviamos mensajes (muro, tweets, ríos de eventos), la forma en la que interactuamos con las aplicaciones (las cuales se vuelven social-aware) y la forma en la que interactuamos con los contenidos de la web, socializándolos (comentar, compartir, valorar, taggear).

Actualmente, existe un pequeño (aunque creciente) conjunto de estrategias reutilizables para socializar contenidos (contenidos publicados fuera de los SNS). SNS como Facebook, Twitter y MySpace proveen widgets para incluirlos manualmente durante la publicación de contenidos.



Ilustración 1: Página de Greenpeace enriquecida con widgets

Cada widget provee un mecanismo o un aspecto diferente de interacción con la red social destino. Por ejemplo el botón 'Like en Facebook' (que puede apreciarse en la Ilustración 1) permite al usuario expresar su interés en el recurso e indica cuantos usuarios lo han hecho. Así, un sitio web enriquecido socialmente, es un sitio que publica contenidos y provee (integra) mecanismos para vincularlos con las redes sociales. No se trata de un mecanismo aislado, ya que W3C incluye un conjunto de publicaciones recientes referentes al fenómeno de widgets recién mencionado, dentro del Web Application Working Group [12]. Ciertas plataformas para la publicación de contenidos, como Drupal y Joomla, simplifican la inclusión de estos widgets al mismo tiempo que permiten combinarlos con versiones propias de los mismos. Algunos SNS proveen extensiones para los navegadores web que permiten socializar contenidos a medida que se los navega sin que ellos hayan sido preparados para tal fin. El plugin de del.icio.us [1], permite agregar un recurso web a la lista de bookmarks del usuario, comentarlo, catalogarlo, valorarlo y compartirlo con el mundo y la red social.

Si bien las estrategias y tecnologías existentes allanan el camino a la socialización de contenidos, tienen limitaciones. En su mayoría son cerradas, puesto que no pueden configurarse ni extenderse. Determinan o fijan la red social a la que referirán las acciones de socialización y en el caso de que los widgets deban agregarse manualmente, ensucian los contenidos. Por otro lado, están disponibles sólo en ciertas plataformas de generación de contenidos y siguen distintas filosofías, lo cual implica que son difícilmente integrables e intercambiables. Finalmente, mezclan (en un único artefacto de software) aspectos como acción, red destino y presentación gráfica.

La WWW dejó de ser una red de páginas HTML y archivos multimedia para convertirse en una vasta base de conocimiento que es transformado, enriquecido, consumido y presentado de diversas

maneras. Este fenómeno se refleja en iniciativas como la de Web Semántica [10], los Web Mashups [3], Linking Open Data [9] y la actividad reciente de la W3C vinculada al desarrollo de especificaciones para la web social federada [13]. En ese contexto, el objetivo del proyecto SocialPipes es potenciar la generación de sitios enriquecidos socialmente. Nuestra hipótesis de trabajo es que se puede aumentar el impacto de las SNS si se simplifica la inserción de funcionalidad de redes sociales en sitios existentes de contenidos. Un punto fundamental de esa simplificación es proveer herramientas que automatizan el enriquecimiento de sitios, aun combinando funcionalidad de distintas SNS, escondiendo la complejidad de esta tarea. En particular, nos interesa un tipo especial de interacción que permite consumir contenidos independientemente de la forma en que son visualizados.

2. EL RETO DE LA SOCIALIZACIÓN NO INTRUSIVA

Un sitio web enriquecido socialmente está compuesto por dos partes principales: contenido original y contenido social. El primero está expresado por código HTML que despliega la información dispuesta por el sitio. El segundo está representado por widgets de proveedores sociales.

Dar soporte para enriquecer sitios de contenidos es un reto con múltiples facetas. Se debe permitir la integración con distintas redes, ya sea para llegar a más usuarios, como así también para incorporar distintos aspectos de socialización. Esto, por ejemplo, implicaría repetir el widget relacionado al aspecto social requerido (por ejemplo, el widget para comentar), para cada proveedor social al cual deseamos integrarnos. Los widgets que actualmente ponen a disposición los sitios de redes sociales para ser incorporados en sitios de terceros resultan muy poco editables. Esto impacta directamente en la vista final del sitio, teniendo que adaptarse a dichos widgets o convivir con esta diferencia entre estilos.

Cada SNS define su propia API (conjunto de funciones que pueden ser utilizadas desde otro sistema) de acuerdo a los aspectos que fueron tenidos en cuenta al momento de la concepción, tanto de dicha API, como de la red social en sí misma. Entonces, cada SNS especifica el modo en que debe ser utilizada su API para consumir contenidos sociales, establece un protocolo de comunicación server-to-server y define diferentes mecanismos para la integración con un sitio arbitrario. Normalmente, existen librerías en diferentes lenguajes de programación, que permiten la interacción server-to-server con el proveedor. Esto impone restricciones con respecto al lenguaje de programación a utilizar, existiendo la posibilidad de que un sitio de red social no provea librerías en determinados lenguajes. Para casos en los que se quiere enriquecer sitios utilizando una arquitectura server-to-server, los mecanismos de autenticación pueden variar de acuerdo a la red social que utilizemos. Existen protocolos de autenticación estándares, como OAuth 1.0 y 2.0 [11], y aproximaciones propias de cada SNS, como Facebook for Websites [8]. Los aspectos relacionados con la interacción server-to-server con un sitio de red social en particular deberán quedar encapsulados, exponiendo solo el punto de conexión entre el aspecto social y la red elegida. En este marco es también importante la independencia de tecnologías de programación puntuales. El mecanismo que predomina entre los proveedores de servicios y aspectos sociales son los llamados a operaciones REST para el manejo de información.

Es necesario partir de la separación del contenido de la presentación dado que nos da la posibilidad de proveer servicios de

publicación que pueden ser consumidos por otros servicios con diferentes objetivos, por ejemplo para ser integrado en un Web Mashup. Con el objetivo de minimizar el esfuerzo de publicación y mantenimiento, se requiere una estrategia no intrusiva, que evite la alteración permanente del contenido original.

En un contexto tan dinámico como la web actual, se debe partir de un diseño flexible, que permita incorporar a nuevos proveedores de aspectos sociales. El diseño debe ser acompañado de herramientas de alto nivel que minimicen la necesidad de programar, permitiendo especificar configuraciones de modo interactivo y sencillo.

3. SOCIAL PIPES: INCLUSIÓN NO INTRUSIVA DE ASPECTOS DE SNS

Social Pipes es un servicio "Enriquecedor social de sitios" compuesto, como se muestra en la Ilustración 2, por un conjunto de componentes conectados en una estructura tipo tubería (pipeline). Cada componente especifica un conjunto de cambios a ser aplicados, toma la salida del componente anterior, y alimenta al siguiente.

El *Generador de Contenidos* publica los contenidos originales como documentos XML, haciendo explícita la semántica de los mismos en el esquema correspondiente. Permite la definición de contenidos independientemente de la forma en la que serán visualizados y socializados. El *Socializador* permite la especificación e inyección de la socialización a aplicar. Dicha especificación se compone de dos partes fundamentales: los aspectos de socialización (p.e., comentar, gustar, compartir) y la implementación de los mecanismos de socialización (en Facebook, en Twitter, etc.). Por último, el *Visualizador* determina cómo serán mostrados los contenidos ya socializados.

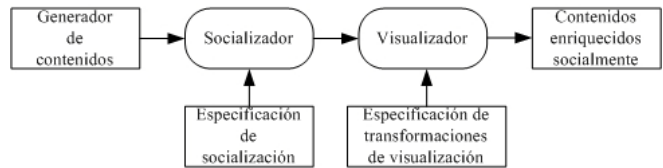


Ilustración 2: Arquitectura general

El Generador de Contenidos (Ilustración 3) atiende solicitudes REST que dan acceso a los distintos elementos del contenido XML (p.e., '/noticias/actual/resumen') y cuyas respuestas son documentos XML. Estos datos podrán ser consumidos por diferentes tipos de servicios web.

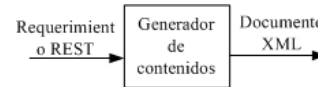


Ilustración 3: Generador de contenidos

A continuación, el servicio Socializador implementa la misma API REST que el generador de contenidos. Los requerimientos al Socializador son reenviados al Generador de contenidos para que dicha respuesta sea extendida con el contenido social. El socializador (Ilustración 4) toma como insumo una especificación de socialización, que es información expresada por medio de un lenguaje construido sobre XML y está enfocada en el XML Schema que nos provee el Generador de Contenidos (semántica explícita). Esta especificación proporciona información sobre los aspectos sociales empleados y las alternativas de implementación utilizadas de los mismos. Además, como el XML Schema actúa

como el 'tipo de dato' de los recursos XML a socializar, dicha especificación no depende del contenido al que se aplique (es fija). A su vez, dicha especificación puede evolucionar (por ejemplo, socializar el mismo contenido con acciones diferentes), lo que significa que es dinámica.

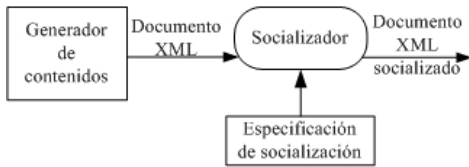


Ilustración 4: Socializador

La componente de Visualización consume la API REST del Socializador y publica una API REST equivalente. El Visualizador reenvía cada la solicitud al Socializador, quien realiza tanto la tarea de comunicación con el Generador de contenidos, así como la transformación para retornar al Visualizador el documento XML socializado. Ya con el recurso socializado, el Visualizador se encarga de transformarlo por medio de una plantilla XSLT y hojas de estilo CSS para la organización y estilos de la presentación, construidas en función de widgets que permiten la interacción social.

El diseño de alto nivel del Socializador (ver Ilustración 5) incluye dos subcomponentes que operan sobre la especificación: un *tejedor de aspectos sociales* y un *tejedor de implementación*. El tejedor de aspectos inyecta los elementos de socialización en el documento XML. Esto incluye los elementos que permiten la acción social (por ejemplo, los que luego resultarán en el botón 'comentar') y los que se utilizan para obtener y desplegar los datos que las acciones han generado (para construir la lista de comentarios asociadas a un artículo). Separa la especificación de los aspectos de socialización y permite la evolución de un "lenguaje de especificación de aspectos sociales" con independencia de los contenidos a ser socializados y de la forma en que serán implementados. A continuación, el tejedor de implementaciones toma 'especificación de implementación social', interactúa con los SNS correspondientes e inyecta en el documento los artefactos sociales correspondientes. Este punto permite el uso intercambiable y combinado de distintas alternativas de implementación de los aspectos de socialización. Incluso, permite la integración de los contenidos en distintas plataformas de socialización, así como la evolución independiente de las implementaciones de aspectos sociales.

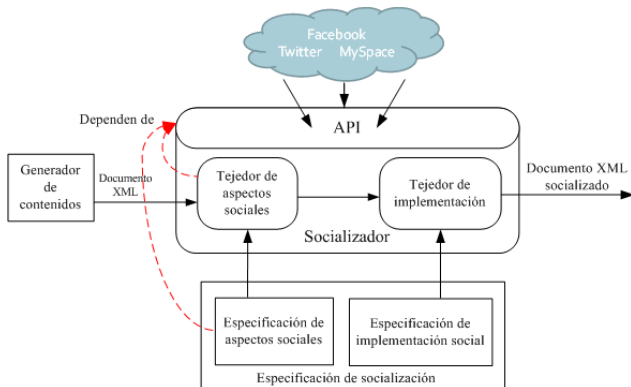


Ilustración 5: Diseño del componente Socializador

El Socializador ve a todas las redes sociales en una misma API. El objetivo fundamental de su definición es capturar aquellas

acciones sociales más interesantes para el fenómeno de enriquecimiento de sitios y abstraernos de los métodos particulares a los servicios sociales. Actualmente, las alternativas de aspectos sociales que forman parte de esa API son 'comentar', 'compartir' y 'calificar/gustar'. Tanto el tejedor de aspectos sociales como el lenguaje de especificación de socialización (aspectos sociales) dependen fuertemente de dicha API, puesto que la evolución de la misma implicará realizar cambios en ambos subcomponentes.

En resumen, el servicio de socialización es una puerta de enlace a la información e interacción social, referenciando el contenido desde el Generador de contenidos. Esto significa que el servicio no almacena información social, ni implementa la funcionalidad de socialización sino que delega esta responsabilidad en un servicio social externo, por ejemplo, Facebook. Es importante aclarar que nuestra herramienta no contempla funcionalidad vinculada a acciones como la adición de contactos, adición de contenidos o control de privacidad, ya que por ejemplo las APIs de los SNS considerados no dan soporte para el manejo de usuarios y, en el caso de hacerlo, no es clara su utilidad en SocialPipes para contemplar los objetivos que nos hemos propuesto. En cuanto al control de privacidad, cada usuario accede al documento enriquecido en el contexto de las redes sociales en las que está identificado. Por tanto, el esquema de privacidad y seguridad se hereda del proveedor social.

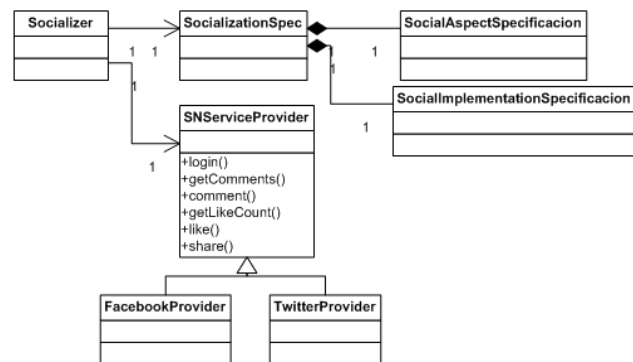


Ilustración 6: API de SNS como punto de extensión

El Socializador es el componente central de la arquitectura. Su estructura de diseño tipo framework permite que el mismo sea extendido para agregar nuevos proveedores de redes sociales con facilidad. En el diseño general que se muestra en la Ilustración 6, la API representa el punto de extensión del cual se derivarán las distintas implementaciones. El diseño general está expresado mediante el paradigma de programación orientada a objetos y contempla la utilización de los conceptos del patrón Puente (o Bridge) [5]. En base a estos, hemos buscado desacoplar nuestra especificación de API Social de los distintos proveedores sociales donde se pueden implementar. Además, el conjunto de métodos que implementan la API Social siguen la estructuración indicada por el patrón Método Plantilla (o Template Method) [5].

Para llevar a cabo la identificación y autorización de usuarios es necesario un mecanismo genérico y que permita conseguir extensibilidad. Por tanto, se optó por la utilización del estándar OAuth [11].

Si se desea incluir un nuevo proveedor de servicios de redes sociales, el mismo debe cumplir un conjunto de requisitos para poder interactuar con el Socializador. En primer lugar, debe disponer de un mecanismo de integración que permita consumir sus

contenidos desde un servidor externo. En ese caso, se trata de aplicaciones que los usuarios de cada red pueden crear. En segundo lugar, que provea un mecanismo de interacción servidor-a-servidor orientado a solicitudes REST. Finalmente, el proveedor social debe soportar e implementar el mecanismo de autenticación delegado OAuth (1.0 o 2.0). Teniendo en cuenta lo anterior, para la inclusión de un nuevo proveedor debemos en primera instancia crear una aplicación en el proveedor social. En segundo lugar, debemos realizar algunos ajustes en la herramienta de socialización. Este paso consiste en agregar un archivo de propiedades que contendrá información vinculada al proveedor que se agrega (parámetros de la aplicación en el proveedor social y el nombre de la clase en nuestro modelo). Finalmente, debemos modificar el modelo para incluir una clase que represente al proveedor e implemente aquellos aspectos específicos a él. La misma deberá extender la superclase 'SNSServiceProvider' e implementar los métodos abstractos correspondientes a los aspectos definidos en esta.

4. CONCLUSIONES Y TRABAJOS FUTUROS

La plataforma descrita fue implementada en Java y se evaluó en la socialización de un servicio de newsletter corporativo, vinculándolo a distintas combinaciones de acciones de socialización de Facebook y Twitter. La evaluación permitió observar que estrategia de socialización no intrusiva encadenando servicios es aplicable y reduce el esfuerzo de socialización de contenidos web.

La API definida provee un enfoque abstracto para abordar el problema de cuáles son los aspectos valiosos para la interacción social y es necesaria para abstraernos de los proveedores sociales. La arquitectura definida mediante la integración como servicios web independientes permite la evolución en forma separada de cada componente. Sin embargo, el constante cambio de las APIs de los SNS existentes (producto de la inmadurez de las mismas) y de la falta de homogeneidad entre proveedores, hacen que la implementación y mantenimiento de los "puentes" de la API sea costosa. Si las APIs de nuestros proveedores sociales cambian, debemos ajustar las clases concretas correspondientes cada uno en la jerarquía 'SNSServiceProvider'. Por lo tanto, nuestro modelo de solución es simple en gran medida por una limitación de los proveedores sociales actuales. Nuestro objetivo no es dar un amplio soporte a los SNS existentes, sino proveer una herramienta extensible para el enriquecimiento. En esencia, hemos desarrollado un framework para la socialización de contenidos. La funcionalidad considerada y los proveedores sociales estudiados fueron suficientes para concebir una prueba de concepto del mismo. Será necesario analizar hasta qué punto es factible la incorporación de otras acciones (y los respectivos servicios que las implementen).

La esencia del problema atacado por SocialPipes es la separación de múltiples aspectos: el contenido, su socialización y su visualización. En los sitios de contenidos web sociales que vemos en la actualidad estos tres aspectos están fuertemente acoplados. Introducir modificaciones a uno cualquiera de ellos (por ejemplo, que objetos socializar) conlleva el riesgo de introducir errores en los otros. En ingeniería de software se conoce este problema como separación de concerns y para él existen métodos como la programación orientada a aspectos (AOP) y tecnologías asociadas [7].

En el trabajo 'Fostering Groupware Tailorability Through Separation of Concerns' [6] hemos discutido como la separación

avanzada de concerns se aplica al dominio de los sistemas de groupware. Dado que el software de redes sociales es un tipo de groupware, es parte de nuestra agenda futura relacionar ambas líneas de trabajo. Es importante aclarar que los preceptos de AOP se aplicaron conceptualmente. Nos enfocamos en ellos para llevar a cabo el proceso de abstracción, definición e identificación del conjunto estándar de acciones, encontrando una analogía directa entre el concepto de 'aspecto cruzado' y las operaciones que están presentes en los SNS. En consecuencia, se trabaja sobre el XML producido por el servicio, por eso es que las herramientas AOP existentes sobre código Java no aplican. Finalmente, la siguiente etapa en el proyecto consiste en adecuar la estrategia y su implementación a los estándares de la web semántica, teniendo en cuenta que el generador de servicios proveerá datos RDF/OWL [10] y que esas mismas tecnologías son las que debe utilizarse para modelar los aspectos de socialización y la interacción con los proveedores específicos de SNS.

5. REFERENCIAS

- [1] Del.icio.us buttons - <https://addons.mozilla.org/en-us/firefox/addon/delicious-buttons/>
- [2] Leonard Richardson, Sam Ruby. RESTful Web Services. O'Reilly, 2007. ISBN-10: 0-596-52926-0.
- [3] Agnes Koschmider, Victoria Torres, Vicente Pelechano. Elucidating the Mashup Hype: Definition, Challenges, Methodical Guide and Tools for Mashups.
- [4] Web Semántica - http://en.wikipedia.org/wiki/Semantic_Web
- [5] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Patrones de Diseño. Pearson Educación, S.A. Madrid. 2003
- [6] Diego Torres, Alejandro Fernández, Gustavo Rossi, Silvia E. Gordillo: Fostering Groupware Tailorability Through Separation of Concerns. CRIWG 2007: 143-156
- [7] Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J.-M., Irwin, J.: Aspect-oriented programming. In: Aksit, M., Matsuoka, S. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 220–242. Springer, Heidelberg (1997)
- [8] Facebook for Websites - <http://developers.facebook.com/docs/guides/web/>
- [9] Linking Open Data Project - <http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>
- [10] Grigoris Antoniou and Frank van Harmelen. 2008. A Semantic Web Primer, 2nd Edition (Cooperative Information Systems) (2 ed.). The MIT Press.
- [11] OAuth - <http://oauth.net/>
- [12] W3C Web Applications (WebApps) Working Group - <http://www.w3.org/2008/webapps/>
- [13] W3C Federated Social Web Incubator Group - <http://www.w3.org/2005/Incubator/federatedsocialweb/>